

SOLUTIONS OF TWISTED WORD EQUATIONS, EDT0L LANGUAGES, AND CONTEXT-FREE GROUPS

VOLKER DIEKERT AND MURRAY ELDER

ABSTRACT. We prove that the set of all solutions for twisted word equations with regular constraints is an EDT0L language and can be computed in PSPACE. It follows that the set of solutions to equations with rational constraints in a context-free group (= finitely generated virtually free group) in reduced normal forms is EDT0L. We can also decide (in PSPACE) whether or not the solution set is finite, which was an open problem. Our results generalize the work by Lohrey and Sénizergues (ICALP 2006) and Dahmani and Guirardel (J. of Topology 2010) with respect to complexity and with respect to expressive power. Neither paper gave any concrete complexity bound and both rely on the exponent of periodicity, so the result in these papers concern only subsets of solutions, whereas our results concern all solutions. We do more, we give, in some sense, the “optimal” formal language characterization of the full solution set.

1. INTRODUCTION

In a seminal paper [24] Makanin showed that the problem *WordEquations* is decidable. The first complexity estimation of that problem was a tower of several exponential functions, but this dropped down to PSPACE by Plandowski [29] using compression. The insight that long solutions of word equations can be efficiently compressed is due to [30] which also led to the (still valid) conjecture that WordEquations is NP-complete. Until 2013 the known decidability proofs for solving word equations were long and technical with an accompanied reputation for being difficult. This changed drastically when Jež applied his *recompression* technique: he presented a simple $\text{NSPACE}(n \log n)$ algorithm to solve word equations [18]. Actually his method achieved more: it describes all solutions, copes with rational constraints (which is essential in applications), and it extends to free groups [6]. Built on the ideas in [6] Ciobanu and the present authors showed that the full solution set of a given word equation with rational constraints is EDT0L [3]. (This was known before only for quadratic word equations by [11].) *EDT0L-languages* are defined by a certain type of Lindenmayer systems, see [35]. The meaning EDT0L is not easy to initially digest, but fortunately there is a very convenient characterization of an EDT0L-language: all we need is a rational set of endomorphisms over a free monoid which applied to any word yields the language. The motivation for [3] was to prove that the full solution set in reduced words of equations in free groups is an indexed language, a problem which was open at that time [12, 17]. However, the result is stronger since EDT0L forms a strict subclass of indexed languages [9].

2010 *Mathematics Subject Classification.* 03D05, 20F65, 20F70, 68Q25, 68Q45.

Key words and phrases. Equation in a virtually free group; twisted equation; EDT0L language; PSPACE. .

Research supported by the Australian Research Council (Discovery Project DP160100486).

Transfer results as in [15, 3] from words to free groups have a long history. In the 1980s Makanin showed that the existential and positive theories of free groups are decidable [25]. In 1987 Razborov gave a description of all solutions for an equation in a free group via “Makanin-Razborov” diagrams [31, 32] which formed a cornerstone in the independent work of Kharlampovich and Myasnikov [20] and Sela [37] on the positive solution of Tarski’s conjectures about the elementary theory in free groups.

The motivation for the present paper is along this line. We show that given a finitely generated (f.g. for short) virtually free group V there is a PSPACE-algorithm which produces for a given equation with rational constraints an effective description of EDTOL languages which describes the full solution set in reduced word over a natural set of generators. Several remarks are in order here. First, in general virtually free groups have torsion; and this is a serious obstacle to apply the known techniques. The reason to study virtually free groups is motivated by ubiquitous presence of *word hyperbolic groups* [14]. Solving equations in torsion-free hyperbolic groups reduces to solving equations in free groups [34], but solving equations in word hyperbolic groups with torsion reduces to solving equations in virtually free groups which in turn reduces to solving “twisted” word equations with rational constraints [4]. The question of whether solving “twisted” word equations is decidable was asked by Makanin ([26] Problem 10.26(b)) and solved in [4]. The conclusion was that it is decidable whether a given equation over a f.g. virtually free group is solvable. This result which was also independently shown by Lohrey and Sénizergues [23]. (Actually, [23] proves a more general transfer result.) What is in common: both papers use a bound on the so-called “exponent of periodicity”. Because of this both are unable to describe all solutions. Moreover, neither paper gives any concrete complexity bounds.

Therefore, the present paper extends [4, 23] in various aspects. For a fixed f.g. virtually free group V we present a PSPACE-algorithm where on input an equation with rational constraints the output is an NFA which defines the full solution set as an EDTOL language. This is done by solving the same problem for twisted word equations with rational constraints. So, the main new contribution is entirely within combinatorics on words. We must deal with twisted word equations; and although we follow the general scheme to define a sound and complete algorithm to produce an NFA \mathcal{A} describing all solutions, the technical details are quite far from previous methods. For example, for twisted equations it does not make sense to “uncross” pairs ab where a, b are different letters because once all pairs ab are uncrossed the twisting may produce new crossing pairs ba , uncrossing them leads to new crossing pairs ab etc. Thus, our underlying method is quite different from the original recompression due to Jež.

The class of f.g. virtually free groups appears in many different ways. For example, a fundamental theorem of Muller and Schupp (relying on [8]) says that f.g. group is virtually free if and only if it is *context-free* [27]. This means that, given any set of monoid generators A , the set of words $w \in A^*$ which represent $1 \in V$ forms a context-free language. Other characterizations include: (1) fundamental groups of finite graphs of finite groups [19], (2) f.g. groups having a Cayley graph with finite treewidth [22], (3) universal groups of finite pregroups [33], (4) groups having a finite presentation by some geodesic string rewriting system [13],

and (5) f.g. groups having a Cayley graph with decidable monadic second-order theory [22]. Proofs for these equivalences are in [7]. The transformations are effective. For example, starting from a context-free grammar for the word problem, we can construct a representation as a fundamental group of finite graphs of finite groups. Then we can find a representation as a free-by-finite group. (We repeat a proof for that step here because we need it for the complexity bounds.) Having that, it is immediate to construct a (deterministic) push-down automaton for the word problem, which yields some context-free grammar for the word problem. Thus, it is legitimate to assume our input is a fundamental group of a finite graph of finite groups with its natural set of generators.

2. PRELIMINARIES

By an *alphabet* we mean a finite set. The elements are called *letters* (or *symbols*). By Σ^* we denote the free monoid over Σ . The elements of a free monoid are called *words*. The length of word w is denoted by $|w|$, and $|w|_a$ counts how often a letter a appears in w . Let M be any monoid and $u, v \in M$. We say that u is a *factor* of v , if we can write $v = xuy$ for some $x, y \in M$. If we can write $v = uy$ (resp. $v = xu$), then we say that u is a *prefix* (resp. *suffix*). If u is a prefix of v , then we also write $u \leq v$. The prefix relation is of no interest in groups, of course. With a few exceptions we denote the neutral element in monoids by 1. In particular, the empty word in a free monoid is also denoted by 1. An *involution* of a set is a bijection $x \mapsto \bar{x}$ such that $\bar{\bar{x}} = x$ for all x in the set. For example, the identity map is an involution. A *monoid with involution* additionally has to satisfy $\overline{xy} = \bar{y}\bar{x}$. If G is a group, then it is a monoid with involution by taking $\bar{g} = g^{-1}$ for all $g \in G$. By default, we identify \bar{g} and g^{-1} for group elements. In the following, every alphabet comes with an involution.

A *morphism* between sets with involution is a mapping respecting the involution. A *morphism* between monoids with involution is a homomorphism $\varphi : M \rightarrow N$ such that $\varphi(\bar{x}) = \bar{\varphi(x)}$. For $\Delta \subseteq M \cap N$ we say that it is a Δ -*morphism* if $\varphi(x) = x$ for all $x \in \Delta$. A bijective morphism is called an *automorphism* and the set of automorphisms on a set (or monoid) M forms the group $\text{Aut}(M)$.

Every group homomorphism is a morphism of monoids with involution. Let G be a group. It acts on a set (with involution) X by a mapping $x \mapsto g \cdot x$ if $1 \cdot x = x$, $f \cdot (g \cdot x) = (fg) \cdot x$ (and $f \cdot \bar{x} = \bar{f \cdot x}$) for all $f, g \in G$ and $x \in X$. If G acts on a monoid (with involution) M , then we additionally demand that every group element acts as an automorphism: $f \cdot (xy) = (f \cdot x)(f \cdot y)$. For better readability we frequently write $f(x)$ instead of $f \cdot x$.

The specification of regular constraints is given here by assigning to each constant and variable an element in finite monoid (typically the finite monoid is a monoid of Boolean matrices and arises as the transformation monoid of a finite automaton.) By making the finite monoid larger, we can turn it into a monoid N with involution and where G acts on it, see the appendix. This allows us to represent regular constraints using a morphism $\mu : (A \cup (G \times \mathcal{X}))^* \rightarrow N$ which respects the involution and the action of G . In the following we fix the finite monoid N and we assume that all morphisms to N respect the involution and G action. We say that M is a *NG-i-monoid* if M is a monoid with involution and a G action together with a morphism $\mu : M \rightarrow N$. If not explicitly stated otherwise all monoids under consideration are

NG -i-monoids. In particular, N is an NG -i-monoid. A morphism between NG -i-monoids is morphism $\varphi : M \rightarrow M'$ such that $\varphi(g \cdot x) = g \cdot (\varphi(x))$ and $\mu' \varphi = \mu$. In the following, if not explicitly stated otherwise a morphism means a morphism between NG -i-monoids.

The guiding example is given as follows. Let A be an alphabet with involution. (Taking the identity means that our results also apply to the case of free monoids without a predefined involution; another typical situation is a subset of a group and $\bar{g} = g^{-1}$.) The involution extends to A^* : for a word $w = a_1 \cdots a_m$ we let $\bar{w} = \bar{a}_m \cdots \bar{a}_1$. The monoid A^* is called the *free monoid with involution over A* . If $\bar{a} = a$ for all $a \in \Sigma$ then \bar{w} is simply the word w read from right-to-left and $\bar{w} = w$ means that w is a palindrome. Note that every automorphism on A^* is induced by some automorphism on the unique minimal generating set A . Thus, $\text{Aut}(A) = \text{Aut}(A^*)$ and $|\text{Aut}(A^*)| \leq |A|!$. In our setting G is a finite subgroup of $\text{Aut}(A)$: the action of G on words is length preserving and it respects the involution. We allow self-involuting letters $a = \bar{a}$. As a consequence we also allow that there are $f \in G$ and $a \in A$ such that $f(a) = \bar{a}$. It is also clear that $f(uv) = f(u)f(v)$ for $f \in G$ and $u, v \in A^*$. If $a \in A$ is a letter and $w \in A^*$ is a word, then we say that a is *G -visible in w* if $f(a)$ appears in w for some $f \in G$. Moreover, assume that for each letter $a \in A$ we are given $\mu(a) \in N$ such that $\overline{\mu(a)} = \mu(\bar{a})$ and $f(\mu(a)) = \mu(f(a))$ for all $f \in G$. Then we can extend μ to the free monoid with involution such that A^* becomes an NG -i-monoid. We also say that A is an *NG -i-alphabet*.

2.1. Rational languages and EDTOL. *Regular languages* in finitely generated free monoids can be defined via *nondeterministic finite automata* (NFA for short) or via recognizability via homomorphisms to finite monoids, to mention just two possible definitions. The notion of a nondeterministic finite automaton extends to every monoid M as follows. An NFA is a directed finite graph \mathcal{A} with initial and final *states*, where the transitions are labeled with elements of the monoid M . A transition labeled by $1 \in M$ is called an ε -*transition* as it is the tradition for NFA's over free monoids. We say that $m \in M$ is *accepted* by the automaton \mathcal{A} if there exists a path from some initial to some final state such that multiplying the edge labels together yields m . This defines the accepted language $L(\mathcal{A}) = \{m \in M \mid m \text{ is accepted by } \mathcal{A}\}$. According to [10] a subset $L \subseteq M$ is *rational* if and only if L is accepted by some NFA over M .

An NFA is called *trim* if every state is on some path from an initial to a final state. For a trim \mathcal{A} we have $L(\mathcal{A}) \neq \emptyset$ if and only if $\mathcal{A} \neq \emptyset$. The acronym EDTOL refers to **E**xtended, **D**eterministic, **T**able, **O** interaction, and **L**indenmayer, see the handbook [36]. A subset L in a k -fold direct product $A^* \times \cdots \times A^*$ is called *EDTOL* if there some (extended) alphabet C with $c_1, \dots, c_k \in C$ such that $A \subseteq C$ and a rational set $\mathcal{R} \subseteq \text{End}(C^*)$ of endomorphisms over C^* such that

$$L = \{(h(c_1), \dots, h(c_k)) \mid h \in \mathcal{R}\}.$$

This means that we have an effective description of L as follows: there is an NFA \mathcal{A} where the transitions are labeled by a “deterministic table” of pairs $(c, u_c) \in C \times C^*$ (encoding the endomorphism which maps c to u_c) and there are letters $c_1, \dots, c_k \in C$ such that $(w_1, \dots, w_k) \in L$ if and only if there is some $h \in L(\mathcal{A}) \subseteq \text{End}(C^*)$ such that $(w_1, \dots, w_k) = (h(c_1), \dots, h(c_k))$. The idea is that properties of L (like emptiness or finiteness) become structural properties of \mathcal{A} . The classical situation

refers to $k = 1$; and our definition uses a characterization of EDT0L languages due to Asveld [1].

2.2. Twisted variables and NG -i-monoids with types. Let B and \mathcal{V} be two disjoint NG -i-alphabets. We call B the alphabet of *constants* and \mathcal{V} the set of *twisted variables*. It is convenient to choose a set \mathcal{X} with involution of minimal size such that every $Y \in \mathcal{V}$ has the form $Y = f(X)$ for some $X \in \mathcal{X}$ and $f \in G$. In the following, by a *variable* we mean $X \in \mathcal{X}$ and thus, every twisted variable $Y \in \mathcal{V}$ can be written as $f \cdot X$ for some $f \in G$. Moreover, we assume $X \neq \overline{X}$ for all variables. If G acts without fixed points on \mathcal{V} , then we identify $\mathcal{V} = G \times \mathcal{X}$ and the action becomes $g \cdot (f, X) = (gf, X)$. By $M(B, \mathcal{X}, \theta, \mu)$ we denote an NG -i-monoid which is generated by $B \cup \{f(X) \mid f \in G, X \in \mathcal{X}\}$ together with a finite set θ of *homogeneous* defining relations. That is, every $(x, y) \in \theta$ satisfies $|x| = |y|$. We always assume that $(x, y) \in \theta$ implies $\mu(x) = \mu(y)$, $(\overline{x}, \overline{y}) \in \theta$, and $(f(x), f(y)) \in \theta$ for all $f \in G$, even if these relations are not listed in the specification of θ .

For complexity issues we require $|x| \leq 2$ for each $(x, y) \in \theta$ and $|\theta| \in \mathcal{O}(|G| \|\mathcal{S}\|^2)$ where $\|\mathcal{S}\|$ is specified in Theorem 2. The homogeneity condition makes it possible to solve the word problem and all other computational issues in the quotient monoid $M(B, \mathcal{X}, \theta, \mu) = M(B, \mathcal{X}, \emptyset, \mu) / \{x = y \mid (x, y) \in \theta\}$. The uniform complexity is in nondeterministic linear space which is good enough for our purposes. These details are easy to see and left to the interested reader. By $M(B, \theta, \mu)$ we denote the NG -i-monoid-submonoid which is generated by B .

3. THE MAIN RESULT ON TWISTED WORD EQUATIONS

We begin with an alphabet of constants A (as always with involution) where G is a subgroup of $\text{Aut}(A)$ as in Section 2. Initially, the set of twisted variables is $G \times \mathcal{V}$ where \mathcal{V} denotes the initial set of variables. The group G acts by $f \cdot (g, X) = (fg, X)$, and hence, without fixed points on twisted variables. For a word w in constants and $f \in G$ we use the notation $f(w) = (f, w)$; and we hence identify $(A \cup (G \times \mathcal{V}))^* = ((G \times (A^* \cup \mathcal{V}))^*)$. We abbreviate $(1, x)$ as x for $x \in A^* \cup \mathcal{V}$. By $\mu_0 : A^* \rightarrow N$ we mean a homomorphism which respects the involution and the action of G . Thus A^* is, via μ_0 , an NG -i-monoid. Assume that μ_0 has been extended to a mapping $\mu_0 : A^* \cup \mathcal{V} \rightarrow N$ such that $\mu_0(\overline{X}) = \overline{\mu_0(X)}$, then μ_0 extends to a morphism $\mu_0 : (A \cup (G \times \mathcal{V}))^* \rightarrow N$ of NG -i-monoids by $\mu_0(f, X) = f \cdot \mu_0(X)$. At this stage we don't have defined types, so we work over free monoids.

A system \mathcal{S} of *twisted word equations with regular constraints* is given by the following data:

- A set of pairs $\{(U_i, V_i) \mid 1 \leq i \leq s\}$ where $U_i, V_i \in (A \cup (G \times \mathcal{V}))^*$ are *twisted words*.
- A morphism $\mu_0 : (A \cup (G \times \mathcal{V}))^* \rightarrow N$.

As usual, a twisted equation (U_i, V_i) is also simply written as $U_i = V_i$. A *solution* of \mathcal{S} is given a morphism $\sigma : \mathcal{V} \rightarrow A^*$ which is (uniquely) extended to an A -morphism of NG -i-monoids $\sigma : (A \cup (G \times \mathcal{V}))^* \rightarrow A^*$ such that

- $\sigma(U_i) = \sigma(V_i)$ for all twisted equations (U_i, V_i) .
- $\mu_0 \sigma(X) = \mu_0(X)$ for all variables. Hence, $\mu_0 \sigma = \mu_0$.

Example 1. Let $A = \{a, \overline{a}, b, \overline{b}\}$, $\mathcal{V} = \{X, \overline{X}, Y, \overline{Y}, Z, \overline{Z}\}$, $f, g \in G$ defined by $f(a) = b, g(a) = \overline{a}, g(b) = b$, $U_1 = (f, X)a(g, \overline{Y})$, $V_1 = Z$, $U_2 = (f, Y)b$, $V_2 = \overline{a}b(g, X)$,

$U_3 = Xa$, $V_3 = b(f, X)$ and (for simplicity) $\mu_0(x) = 1$ for all $x \in A \cup \mathcal{V}$. A solution is given by $\sigma(X) = bab$, $\sigma(Y) = \bar{b}a\bar{a}\bar{b}$, $\sigma(Z) = abaabaab$.

If σ is a solution of \mathcal{S} we also say that σ solves \mathcal{S} , then for $\mathcal{V} = \{X_1, \bar{X}_1, \dots, X_k, \bar{X}_k\}$ the full solution set $\text{Sol}(\mathcal{S})$ of \mathcal{S} is defined as

$$\text{Sol}(\mathcal{S}) = \{(\sigma(X_1), \dots, \sigma(X_k)) \in A^* \times \dots \times A^* \mid \sigma \text{ solves } \mathcal{S}\}.$$

Our main structural result shows that $\text{Sol}(\mathcal{S})$ is effectively EDT0L. Actually, we can compute an effective presentation in polynomial space. In order to measure complexities we need a notion of input size. We define the size $\|\mathcal{S}\|$ by

$$\|\mathcal{S}\| = |G| + |A| + |\mathcal{V}| + s + \sum_{1 \leq i \leq s} |U_i V_i|.$$

Convention. There is an implicit bound $|G| \leq |A|!$. Since G can be much smaller, our complexity bounds take $\|\mathcal{S}\|$ and $|G|$ as parameters into account. For better readability we don't measure N . Therefore, we add the general hypotheses that N is given in such a way that the specification and all necessary computations over N , such as performing a multiplication, computing the involution or the G action, can be done in polynomial space with respect to $\|\mathcal{S}\|$. Thus, in the example of Boolean $m \times m$ matrices, we may allow that m is polynomial in $\|\mathcal{S}\|$ although the size of N becomes 2^{m^2} . That is, we do not ever need to actually store the entire set of matrices N .

Theorem 2. *There is a PSPACE algorithm which takes as input a system of twisted word equations with regular constraints \mathcal{S} and \mathcal{V} as above with input size $\|\mathcal{S}\|$. The output is an extended alphabet C of size $\mathcal{O}(|G|^2 \|\mathcal{S}\|^2)$, letters $c_X \in C$ for each $X \in \mathcal{V}$, and a trimmed NFA \mathcal{A} accepting a rational set of A -morphisms $L(\mathcal{A}) \subseteq \text{End}(C^*)$ such that*

$$(1) \quad \text{Sol}(\mathcal{S}) = \{(h(c_{X_1}), \dots, h(c_{X_k})) \in C^* \times \dots \times C^* \mid h \in L(\mathcal{A})\}.$$

The algorithm stores intermediate equations with a length bound in $\mathcal{O}(|G| \|\mathcal{S}\|^2)$. Moreover, $\text{Sol}(\mathcal{S}) = \emptyset$ if and only if $L(\mathcal{A}) = \emptyset$; and $|\text{Sol}(\mathcal{S})| < \infty$ if and only if \mathcal{A} doesn't contain any directed cycle.

Theorem 2 implies that $\text{Sol}(\mathcal{S})$ is effectively EDT0L, and that we can decide in polynomial space (resp. deterministic exponential time) whether \mathcal{S} is solvable and whether or not there are only finitely many solutions. The problem to decide emptiness of $\text{Sol}(\mathcal{S})$ is known to be PSPACE-hard by [21] because the intersection problem of regular languages is a special case. If the finite monoid N is not part of the input, then the best known lower bound is NP-hardness.

4. TWISTED CONJUGACY AND δ -PERIODIC WORDS

Before we dive into the proof of Theorem 2 we show how to solve a particular kind of a twisted equation: conjugacy. For words $x, y, z \in A^*$ with $1 \neq z$ a standard exercise in combinatorics on words [16] shows:

$$(2) \quad zy = xz \iff \exists r, s \in A^* \exists e \in \mathbb{N} : x = rs \wedge y = sr \wedge z = (rs)^e r.$$

This fact is crucial in Makanin's classical approach [24] to solve (untwisted) word equations. Here, we need a variant of (2) in the twisted environment. We say that words $x, y \in A^*$ are *twisted conjugate* if there are $f, g, h \in G$ and $z \in A^*$ such that

$zg(y) = h(x)f(z)$. We also say that $|x| = |y|$ is the *offset* of the conjugacy. A *twisted conjugacy equation* is a twisted equation of the form

$$(3) \quad Z(g, Y) = (h, X)(f, Z).$$

Proposition 3. *Let σ be a solution of the twisted equation (3) such that the offset $|\sigma(X)|$ satisfies $1 \leq |\sigma(X)| < |\sigma(Z)|$. Then there are words $r \in A^+$, $s \in A^*$ and $e, j \in \mathbb{N}$ with $0 \leq j < |G|$ such that $|rs| = |\sigma(X)|$ and*

$$(4) \quad \sigma(Z) = ((rs)f(rs) \cdots f^{|G|-1}(rs))^e f^0(rs) \cdots f^{j-1}(rs)f^j(r).$$

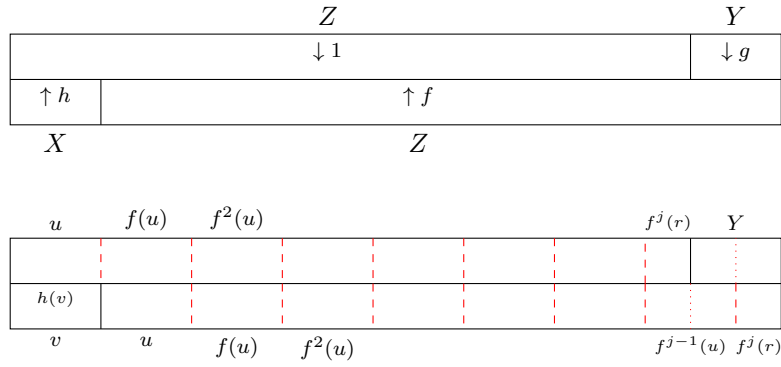


FIGURE 1. Twisted conjugacy

Proof. Let $v = \sigma(X)$ and $u = h(v)$. Since $1 \leq |\sigma(X)| < |\sigma(Z)|$ the word u is a proper nonempty prefix of $\sigma(Z)$. If $2|u| \leq |\sigma(Z)|$, then $uf(u)$ is a prefix of $\sigma(Z)$, and so on. Thus, $\sigma(Z)$ is a prefix of the word in $uf(u)f^2(u) \cdots f^{|\sigma(X)|}(u)$. Next, observe that $f^{|G|}(u) = f^0(u) = u$ for every word $u \in A^*$. Thus,

$$\sigma(Z) = [uf(u)f^2(u) \cdots f^{|G|-1}(u)]^e uf(u) \cdots f^{j-1}(u)f^j(r)$$

where $0 \leq i < |G|$, $u = rs$ and the $|r|$ suffix of Z is where the pattern runs out, see Figure 7. We then have $\sigma(Y) = g^{-1}f^j(sf(r))$. Hence, the nonempty word u and the length $|\sigma(Z)|$ define a unique factorization $u = rs$, integers $0 \leq e$ and $0 \leq j < |G|$ such that $\sigma(Z)$ has the desired form above. \square

Recall that a word p is *primitive* if it cannot be written as $p = r^e$ with $e \geq 2$. In particular, the empty word 1 is not primitive. It is well-known (and easy to see) that a nonempty word p is primitive if and only if p^2 cannot be written as $p^2 = xpy$ with $x \neq 1$ and $y \neq 1$.

Let $w, p \in A^+$ be nonempty words. We say that w has *period* $|p|$ if w is a prefix of $p^{|w|}$. In other words, if $w = a_1 \cdots a_n$ with $a \in A$, then $a_i = a_{i+|p|}$ for all $1 \leq i \leq n - |p|$. A word may have several periods, for example $w = aabaabaa$ has periods 3, 6, 7, 8. If $|p|$ is the least period of w , then $|p| \leq |w|$ and we can choose p to be primitive such that $p \leq w$. For example, $aab \leq aabaabaa$ is a primitive prefix and $|aab| = 3$.

Definition 4. We say that a word w is δ -periodic if it has some period less or equal to than δ . A δ -periodic word w is called long δ -periodic if $|w| \geq 3\delta$, and very long δ -periodic if $|w| \geq 10\delta$.

Using this terminology, Proposition 3 yields the following result.

Corollary 5. Let $\varepsilon \in \mathbb{N}$, $f, g, h \in G$, and $x, y, z \in A^*$ be words with $1 \leq |x| \leq \varepsilon$ and $|z| \geq 10|G|\varepsilon$. If we have $zg(y) = h(x)f(z)$, then z is a very long $|G|\varepsilon$ -periodic word.

Moreover, let $z = \alpha w \beta$ be any factorization with $|w| = |x|$. Then every letter b occurring in z satisfies $b = f(a)$ for some $f \in G$ and some letter a occurring in w .

Proof. By the proposition we have

$$z = ((rs)f(rs) \dots f^{|G|-1}(rs))^e f^0(rs) \dots f^{j-1}(rs)f^j(r)$$

where $|f^i(rs)| = |rs| \leq \varepsilon$ so z has a period

$$|(rs)f(rs) \dots f^{|G|-1}(rs)| \leq |G|\varepsilon,$$

and $|z| \geq 10|G|\varepsilon$ by hypothesis so z is a long $|G|\varepsilon$ -periodic word.

For the second claim, if $z = \alpha w \beta$ with $|w| = |x|$ then w is a factor of $f^i(rs)f^{i+1}(rs)$ of length $|rs|$. If we write $rs = a_1 \dots a_{|x|}$, then any letter b in z satisfies $b = f^k(a_\ell)$. Let $\iota \in \{i, i+1\}$ so that $f^\iota(a_\ell)$ is a letter in v , then $b = f^k(a_\ell) = f^k(f^{-\iota}(f^\iota(a_\ell))) = f^{k-\iota}(a)$. \square

An important property of δ -periodic words is the following.

Lemma 6. Let w be a δ -periodic word and $w = p^e r = q^f s$ such that p, q are primitive $|p| \leq |q| \leq \delta$, $1 \neq r \leq p$, $1 \neq s \leq q$, and $|w| \geq 2\delta$. Then $p = q$, $e = f \geq 1$, and $r = s$.

Proof. The assertion is clear for $|p| = |q|$. Hence we may assume that p is a proper prefix of q . Since $q \leq w$ we conclude $q \leq p^\delta$. Since $|w| \geq 2\delta$, and $|p| \leq |q| \leq \delta$ we see $pq \leq w \leq q^\delta$. Thus q occurs as factor inside qq : we have $pqs = qq$ for some s . Since $1 \leq |p| < |q|$, this contradicts the primitivity of q . \square

Let u be a prefix (resp. factor, resp. suffix) of some nonempty word w . We say that u is a *maximal δ -periodic prefix (resp. factor, resp. suffix)* in w if we cannot extend the occurrence of the factor u inside w by any letter to the right or left, to see a δ -periodic word.

4.1. Preprocessing to pass to a triangulated system. A twisted word equation $U = V$ is called *triangulated* if U contains at most 2 and V at most one variable. By standard methods (using at most $2\|S\|$ more variables), it is enough to show Theorem 2 in the case where each equation $U_i = V_i$ equals $(f, x)(g, y) = (h, z)$ where $x, y, z \in A \cup \mathcal{V}$. This is equivalent to $(h^{-1}f, x)(h^{-1}g, y) = z$. Moreover, we can assume that $z = Z$ is variable. Hence the starting point is a system of equations $(f, x)(g, y) = Z$. During the process we need a more general form, (standard) equations appear as $u(f, x)w(g, y)v = u'Zv'$ where u, w, v, u', v' are words over constants. Whenever such an equation with $|u| = |u'| = |v| = |v'|$ appears, we make a consistency check that $u = u'$ and $v = v'$; in the other case we stop with “unsolvable”. That is, in the nondeterministic process this branch is rejected. Finally, by adding a new zero $0 = \bar{0}$ and a new neutral element $1 = \bar{1}$, to N , we can assume that $\mu(w) = 1$ implies $w = 1$ for all words over constants and $\mu(X) \neq 0$

for variables. Moreover, we assume that for every $u(f, X)w(g, Y)v = u'Zv'$ its dual equation $\overline{v}(g, \overline{Y})\overline{w}(f, \overline{X})\overline{u} = \overline{v'}\overline{Z}\overline{u'}$ is part of the system too.

During the process we have to enlarge the sets of constants and variables. In the beginning we fix two disjoint alphabets with involution C and Ω . All constants are drawn from C and all variables are drawn from Ω . We require $|\Omega| = |C|$ and that $|C|$ is large enough, but polynomial in the input size $\|\mathcal{S}\|$. More precisely we have $|C| \in \mathcal{O}(|G|^2\|\mathcal{S}\|^2)$. Throughout we use following notation.

- $A \subseteq B = \overline{B} \subseteq C$, $\mathcal{X} = \overline{\mathcal{X}}$, $\mathcal{Y} = G \times \mathcal{X} \subseteq \Omega$ and $\Sigma = C \cup \Omega$.
- G acts on $B \cup \mathcal{Y}$, the action and the involution extend those on $A \cup (G \times \mathcal{V})$.
- $\mu : \Sigma^* \rightarrow N$ satisfies $\mu(a) = \mu_0(a)$ for $a \in A$.
- $a, b, c, [p], [r, s, \lambda], \dots$ refer to letters in C .
- u, v, w, \dots refer to words in C^* .
- $X, Y, Z, [X, p], \dots$ refer to variables in Ω , $X \neq \overline{X}$ for all variables.
- x, y, z, \dots refer to words in Σ^* .

These conventions hold everywhere unless explicitly stated otherwise. They also apply to primed symbols such as B' , \mathcal{X}' etc.

4.2. The initial word equation W_{init} . For technical reasons we encode the initial (triangular) system $\{(U_i = V_i) \mid 1 \leq i \leq s\}$ of twisted equations in variables $\mathcal{V} = \{X_i, \overline{X}_i \mid 1 \leq i \leq k\}$ as a single word. For this we assume that A contains special marker symbols $\#$ and $\overline{\#}$ with $\mu(\#) = \mu(\overline{\#}) = 0$. Thus, they cannot be used by any solution $\sigma(X)$ since $\mu(X) \neq 0$ by assumption. We also assume that for each $a \in A$ there is a trivial equation $a = a$. (This ensures that letters from A will always be visible in equations.) Let $U = U_1\# \dots \# U_s\#\overline{U}_1\# \dots \#\overline{U}_s\#$ and $V = V_1\# \dots \# V_s\#\overline{V}_1\# \dots \#\overline{V}_s\#$.

The *initial equation* $W_{\text{init}} \in ((A \cup (G \times \mathcal{V}))^*)$ is defined as:

$$(5) \quad W_{\text{init}} = \#X_1\# \dots \#X_k\#U\#\overline{\#X_1\# \dots \#X_k\#V\#}.$$

Note that $\sigma(W) = \sigma(\overline{W})$ if and only if $\sigma(U_i) = \sigma(V_i)$ for all i . We fix $n = |W_{\text{init}}|$. Note that this implies $n > |A| + |\mathcal{V}|$ and $\|\mathcal{S}\| \in |G| + \Theta(n)$.

Definition 7. An extended equation is a tuple $(W, B, \mathcal{X}, \theta, \mu)$, where θ is a type and:

- (1) $W \in (B \cup \mathcal{Y})^*$ (with $\mathcal{Y} = G \times \mathcal{X}$) and $|W| \leq |C \cup \Omega|$.
- (2) $W = \#x_1\# \dots \#x_k\#u\#\overline{\#x_1\# \dots \#x_k\#v\#}$ for some $u = u_1\# \dots \#u_s\#\overline{u}_1\# \dots \#\overline{u}_s$ and $v = v_1\# \dots \#v_s\#\overline{v}_1\# \dots \#\overline{v}_s$ with $x_i, u_i, v_i \in (B \cup \mathcal{Y})^*$ and $\mu(x_i u_i v_i) \neq 0$.
- (3) Given W as above we call $u_i = v_i$ a local equation.
- (4) We say W as above is a standard equation if $\theta = \emptyset$ and all local equations are triangular
- (5) If a variable X appears somewhere in θ , then X is called typed. We require that for a typed variable X there exists a primitive word $\theta(X) \in B^*$ such that $X\theta(X) = \theta(X)X$ is true in $M(B, \mathcal{X}, \theta, \mu)$.
- (6) A solution is a morphism $\sigma : M(B, \mathcal{X}, \theta, \mu) \rightarrow M(B, \theta, \mu)$ such that:
 - $\sigma(W) = \sigma(\overline{W})$. (Equivalently: $\sigma(u_i) = \sigma(v_i)$ for all i .)
 - $\sigma(X) \in p^*$, whenever X is typed and $p = \theta(X)$.
- (7) An entire solution is a pair (α, σ) where $\alpha : M(B, \theta, \mu) \rightarrow M(A, \emptyset, \mu_0)$ is an A -morphism and σ is a solution.

The states of the NFA \mathcal{A} we are aiming for are extended equations and transitions are certain labeled arcs between states.

If $(W, B, \mathcal{X}, \theta, \mu) \xrightarrow{h} (W', B', \mathcal{X}', \theta', \mu')$ is a transition, then $h : M(B', \theta', \mu') \rightarrow M(B, \theta, \mu)$ is a morphism (in the opposite direction of the arc) which is specified a mapping $h : \Delta \rightarrow B^*$ with $\Delta \cap A = \emptyset$. We assume that it can be extended to a morphism $h : M(B', \theta', \mu') \rightarrow M(B, \theta, \mu)$ by leaving all letters in $B' \setminus \{f(d) \mid d \in \Delta \cup \overline{\Delta}, f \in G\}$ invariant. Thus, if we say that $h(c) = w$, then we don't need to say $h(\bar{c}) = \bar{w}$ or $f(c) = f(w)$ for $f \in G$ or $\mu'(c) = \mu(w)$. Moreover, for each constant $a \in A$ we know its stabilizer $G_a = \{g \in G \mid g(a) = a\} = G_{\bar{a}}$. New constants appear only by compression. This means a word w is replaced a fresh letter c by specifying $h(c) = w$. At this point the stabilizer of w is known: for example if $w = ab$, then $G_w = \{g \in G \mid g(w) = w\} = G_a \cap G_b$. Hence, we define the stabilizer $G_c = G_w$; and we also introduce fresh letters $f(c)$ with the property $f(c) = g(c) \iff g^{-1}f \in G_w$. We also define $\mu(f(c)) = \mu(f(w))$ and $c = \bar{c} \iff w = \bar{w}$; and we check that $h(p) = h(q)$ for all $(p, q) \in \theta$. In this way, we make sure that h is a morphism. (Recall that this means a morphism of NG -i-monoids.) The morphism h induces an endomorphism of C^* which respects the involution assuming $h(c) = c$ for all $c \in C \setminus B'$. However, outside B' neither the action of B nor the value of μ is defined, so C^* is not an NG -i-monoid. The crucial observation is that whenever

$$(W_s, B_s, \mathcal{X}_s, \theta_s, \mu_s) \xrightarrow{h_{s+1}} \dots \xrightarrow{h_t} (W_t, B_t, \mathcal{X}_t, \theta_t, \mu_t)$$

is a labeled path and $w \in B_t^*$ is word, then $h = h_1 \dots h_t$ can be viewed either as a morphism $h : M(B_t, \theta_t, \mu_t) \rightarrow M(B_s, \theta_s, \mu_s)$ or as an endomorphism of C^* . If we have $w \in B_t^*$, then h defines a word $h(w) \in B_s^*$ and the corresponding element $h(w) \in M(B_s, \theta_s, \mu_s)$. By ε we denote the identity of C^* . Then ε appears as a label transitions $(W, B, \mathcal{X}, \theta, \mu) \xrightarrow{\varepsilon} (W', B', \mathcal{X}', \theta', \mu')$ where $h : M(B', \theta', \mu') \rightarrow M(B, \theta, \mu)$ is a morphism with $h(a) = a$ for all $a \in B'$. For example, we might have $B' \subseteq B$ or $(B', \mu') = (B, \mu)$ and $\theta' \subseteq \theta$.

5. THE AMBIENT NFA \mathcal{F}

We are ready to define an NFA \mathcal{F} which contains the trimmed NFA \mathcal{A} we are aiming for as a subautomaton. We show that \mathcal{F} is sound: this means in the notation of Theorem 2

$$(6) \quad \{(h(c_{X_1}), \dots, h(c_{X_k})) \in C^* \times \dots \times C^* \mid h \in L(\mathcal{F})\} \subseteq \text{Sol}(\mathcal{S}).$$

The states of \mathcal{F} are extended equations and there are two types of transitions: a *substitution* transforms the variables and does not affect the constants; a *compression* affects the constants, it may change variables only by changing types.

5.1. States. We define the states of the NFA \mathcal{F} by a subset of extended equations $(W, B, \mathcal{X}, \theta, \mu)$ according to Definition 7, subject to the following *type restriction*: whenever X appears in a defining relation $(x, y) \in \theta$, then (up to symmetry) there is a letter a and a typed variable Y (possibly $X = Y$) such that first, $(x, y) = (Xa, aY)$ and second, $\theta(X)a = a\theta(Y)$ holds in $M(B, \theta, \mu)$. Recall that for a typed variable X there exists a primitive word $\theta(X) \in B^*$ such that $X\theta(X) = \theta(X)X$ holds in $M(B, \mathcal{X}, \theta, \mu)$. (In order to have $\theta(X)$ be defined through θ we might require that $\theta(X)$ is the shortest primitive word with $X\theta(X) = \theta(X)X$ and that it is unique by that property. However, this is not essential.)

Since our monoids are defined through homogeneous relations, type restriction implies $|\theta(X)| = |\theta(Y)|$. Moreover, if $\varphi : M(B, \mathcal{X}, \theta, \mu) \rightarrow M(B', \theta', \mu')$ is any morphism, then $|\varphi(X)| = |\varphi(Y)|$ as well as $|\varphi\theta(X)| = |\varphi\theta(Y)|$.

5.1.1. *Initial state.* The *initial state* is $(W_{\text{init}}, A, \mathcal{V}, \emptyset, \mu_0)$.

5.1.2. *Final states.* We say that $(W, B, \emptyset, \emptyset, \mu)$ is *final* if

- (1) $W = \overline{W}$ and there are no variables.
- (2) The word W has a prefix of the form $\#c_{X_1}\#\cdots\#c_{X_k}\#$.

5.2. Transitions.

5.2.1. *Substitutions.* These are transitions defined by a B -morphism $\tau : M(B, \mathcal{X}, \theta, \mu) \rightarrow M(B, \mathcal{X}', \theta', \mu')$ such that $M(B, \theta, \mu) = M(B, \theta', \mu')$ and whenever $X \in \mathcal{X}$ is typed and $p = \theta(X)$, then $\tau(X) \in p^*Xp^* \cup p^*$ and either $\theta(X) = \theta'(X)$ or $X \notin \mathcal{X}'$. The transition is denoted by $(W, B, \mathcal{X}, \theta, \mu) \xrightarrow{\varepsilon} (\tau(W), B, \mathcal{X}', \theta', \mu')$.

To make the construction effective we also require that $\sum_{X \in \mathcal{X}} |\tau(X)| \leq |C|$.

Lemma 8. *If σ' solves $W' = \tau(W) \in M(B', \mathcal{X}', \theta', \mu')$, then $\sigma = \sigma'\tau$ solves $W \in M(B, \mathcal{X}, \theta, \mu)$. In particular, if $\alpha : M(B, \theta, \mu) \rightarrow M(A, \emptyset, \mu_0)$ is an A -morphism, then (α, σ) and (α, σ') are entire solutions with $\alpha\sigma(W) = \alpha\sigma'(W')$ for $W' = \tau(W)$*

Proof. Since $M(B, \theta, \mu) = M(B, \theta', \mu')$, τ and σ' are B -morphisms. Thus, $\sigma(W) = \sigma'(W') = \sigma'(\overline{W'}) = \sigma(\overline{W})$. Moreover, we know $\tau(X) \in p^*Xp^*$ or $\tau(X) \in p^*$ (possible only if $X \notin \mathcal{X}'$). In the former case we have, by definition, $\sigma(X) = \sigma'\tau(X) \in p^*$ since $p = \theta(X) = \theta'(X)$. The same holds if $\sigma(X) \in p^*$ because σ' leaves constants invariant. \square

5.2.2. *Compressions.* These are transitions defined by a $(A \cup \mathcal{X})$ -morphism $h : M(B', \mathcal{X}, \theta', \mu') \rightarrow M(B, \mathcal{X}, \theta, \mu)$, where the monoids share the same set of variables \mathcal{X} , but the type for variables might be different. Let $E' = (W', B', \mathcal{X}, \theta', \mu')$ be an extended equation and $E = (h(W'), B, \mathcal{X}, \theta, \mu)$. We intend to define a transition denoted by $E \xrightarrow{h} E'$ (pointing to the opposite direction of h). However, the situation is more subtle than for substitutions, so we need restrictions. We assume:

- $\sum_{b' \in B'} |h(b')| \leq |C|$.
- Restriction yields a morphism $h : M(B', \theta', \mu') \rightarrow M(B, \theta, \mu)$.
- If $(Xa, aY) \in \theta$ then $(Xa, aY) \in \theta'$, but we allow $\theta(X) \neq \theta'(X)$.
- If $\theta(X)$ is defined, then $\theta'(X)$ is defined and $h(\theta'(X)) \in \theta(X)^*$.

These assumptions define a transition $(h(W'), B, \mathcal{X}, \theta, \mu) \xrightarrow{h} (W', B', \mathcal{X}, \theta', \mu')$.

Lemma 9. *Let $\sigma' : M(B', \mathcal{X}, \theta', \mu') \rightarrow M(B', \theta', \mu')$ solve W' . Then $\sigma(X) = h\sigma'(X)$ and $\sigma(b) = b$ defines a solution for $h(W')$; and if $\alpha : M(B, \theta, \mu) \rightarrow M(A, \emptyset, \mu_0)$ is an A -morphism, then (α, σ) and $(\alpha h, \sigma')$ are entire solutions with $\alpha\sigma(W) = \alpha h\sigma'(W')$ for $W = h(W')$.*

Proof. By definition, σ leaves letters from B invariant. We have to show that $(x, y) \in \theta$ implies $\sigma(x) = \sigma(y) \in M(B, \theta, \mu)$. This is trivial if neither x nor y contains a variable. In the other case, due to the type restriction above, we have $(x, y) = (Xa = aY)$ for typed variables X, Y and a letter a . We conclude $\theta'(X)a = a\theta'(Y) \in M(B', \theta', \mu')$ and $|\theta'(X)| = |\theta'(Y)|$. Moreover, $(Xa = aY) \in \theta'$ implies $\sigma'(X)a = a\sigma'(Y)$ and $|\sigma'(X)| = |\sigma'(Y)|$. Let $p = \theta(X)$, $p' = \theta'(X)$, $q = \theta(Y)$, and $q' = \theta'(Y)$. Let $\sigma'(X) = p'^\ell$, then $\sigma'(Y) = q'^\ell$, too. By $h(p') \in p^*$

and $h(q') \in q^*$, we deduce from $p'a = aq'$ that $|h(p')| = |h(q')|$. Now let $h(p') = p^k$. Then we can calculate in $M(B, \theta, \mu)$ as follows

$$\sigma(Xa) = h(\sigma'(X))a = h(p')^\ell a = p^{k\ell} a = aq^{k\ell} = \sigma(aY).$$

This tells us that σ is B -morphism and $h\sigma'(x) = \sigma h(x)$ for all $x \in B' \cup \mathcal{X}$. Hence,

$$\sigma(W) = \sigma(h(W')) = h\sigma'(W') = h\sigma'(\overline{W'}) = \sigma(\overline{W}).$$

If $\theta(X)$ is defined, then the above calculation implies $\sigma(X) \in \theta(X)^*$. The assertion $\alpha\sigma(W) = \alpha h\sigma'(W')$ for $W = h(W')$ is trivial. \square

Proposition 10. *Let $E_0 \xrightarrow{h_1} \dots \xrightarrow{h_t} E_t$ be a path in \mathcal{F} of length t , where $E_0 = (W_{\text{init}}, A, \mathcal{V}, \emptyset, \mu_0)$ is an initial and $E_t = (W, B, \emptyset, \emptyset, \mu)$ is a final state. Then E_0 has an entire solution $(\text{id}_{A^*}, \sigma)$ with $\sigma(W_{\text{init}}) = h_1 \dots h_t(W)$. In particular, for $X \in \mathcal{V}$ we have $\sigma(X) = h_1 \dots h_t(c_X)$; and \mathcal{F} is sound in the sense of (6).*

Proof. Since E_t is final, it has a unique solution $\sigma_t = \text{id}_{B^*}$. By the lemmas above, we obtain a solution σ at E_0 such that $\text{id}_{A^*}\sigma(W_{\text{init}}) = \text{id}_{A^*}h_1 \dots h_t \text{id}_{B^*}(W)$. Hence, $(\text{id}_{A^*}, \sigma)$ is an entire solution as desired. \square

6. THE TRIMMED NFA \mathcal{A}

We construct the trimmed NFA \mathcal{A} effectively as a subautomaton in \mathcal{F} . Trimming is not enough because we want to see loops on accepting paths if and only if there are infinitely many solutions. Therefore we first define a *weight* for states.

Definition 11. *Let $E = (W, B, \mathcal{X}, \theta, \mu)$ be an extended equation. The weight $\|E\|$ is a pair of natural numbers $\|E\| = (|W|, |B|)$.*

We order tuples in \mathbb{N}^2 lexicographically, for example $(1, 42) < (2, 0)$; and we use the fact that there are no infinite descending chains in \mathbb{N}^2 .

We now construct \mathcal{A} in two stages. In a first stage we remove all outgoing arcs from final states, but we keep all incoming compression transitions to final states. If E' is not final, then we remove all compression transitions $E \xrightarrow{h} E'$ where $h(a) = 1$ for some letter a or where $\|E\| \leq \|E'\|$. All substitution transitions $E \xrightarrow{\varepsilon} E'$ are defined by some τ . We keep them, only if either $\|E\| > \|E'\|$ or there is some $X \in \mathcal{X}$ such that $1 \leq \sum_{b \in B'} |\tau(X)|_b$. Call the intermediate NFA obtained after this stage \mathcal{F}' .

In a second stage, we remove all states from \mathcal{F}' with adjacent transitions which are not on a path from an initial to a final state. This defines the trimmed automaton \mathcal{A} mentioned in Theorem 2. Standard methods show that \mathcal{A} can be constructed in PSPACE by Savitch's Theorem. Actually, our theorem requires a construction in NSPACE($|G| \|\mathcal{S}\|^2$) and we use the result by Immerman and Szelepcsényi which implies that NSPACE($n \log n$) is (effectively) closed under complementation (see [28, Theorem 7.6]). For more details we refer to [3]. So, complexity issues are not discussed anymore.

Since $\mathcal{A} \subseteq \mathcal{F}$ we obtain (6) for free. Later, we will show that \mathcal{A} is *complete* which is defined as follows.

$$(7) \quad \text{Sol}(\mathcal{S}) \subseteq \{(h(c_1), \dots, h(c_k)) \in C^* \times \dots \times C^* \mid h \in L(\mathcal{A})\}.$$

Proposition 12. *If $\mathcal{A} \neq \emptyset$, then \mathcal{S} has at least one solution. If \mathcal{A} contains a directed cycle, then \mathcal{S} has infinitely many solutions. Moreover, if \mathcal{A} is complete, then the converse of both assertions is true.*

Proof. If $\mathcal{A} \neq \emptyset$, then \mathcal{S} has at least one solution by Proposition 10. Now assume that \mathcal{A} contains a directed cycle, then there is an accepting path with a directed cycle and this cycle doesn't involve any final state as final states are without outgoing arcs. Let $E_s \xrightarrow{h_s} \dots \xrightarrow{h_t} E_t = E_s$ be this cycle. Without restriction we have $t > s$ and $\|E_s\| \leq \|E_{s+1}\|$ because \mathbb{N}^ℓ admits no infinite strictly descending chains. This means $E_s \xrightarrow{\varepsilon} E_{s+1}$ must be a substitution transition. Hence, by definition of \mathcal{A} , it is defined due to some τ with $|\tau(X)|_a \geq 1$ for some X . Hence, on some accepting path we can pop out an arbitrary number of letters of X . Since on paths from an initial state to E_s the labels are nonerasing endomorphisms, we see that we can make $\sigma(X) \in A^*$ at the initial state E_{init} larger and larger. Thus, there are infinitely many solutions. The converse, under the assumption that (7) holds, is trivial. \square

7. COMPLETENESS OF THE NFA \mathcal{A}

By virtue of Proposition 12 we can deduce Theorem 2 if we show that \mathcal{A} is complete. We do so by describing a deterministic algorithm which, on input a fixed solution σ_0 , computes a path labeled by (h_1, \dots, h_t) inside the graph \mathcal{A} from initial to final state $E_t = (W_t, B_t, \emptyset, \emptyset, \mu_t)$ so that $\sigma_0(W_{\text{init}}) = h_1 \dots h_t(W_t)$. Note that this is a purely existential statement, no construction is needed anymore: we simply need to ensure that the path we describe stays inside \mathcal{A} by making constants in \mathcal{O} -terms large enough, and that all transitions we use satisfy the specifications given above.

7.1. Alphabet reduction. Consider a state $E = (W, B, \mathcal{X}, \theta, \mu)$. We need to control the size of B and we therefore wish that it contains only those letters G -visible in W . Thus, we let

$$B' = \{a \in B \mid \exists f \in G : |W|_{f(a)} \geq 1\}$$

and θ', μ' be the restrictions of θ and μ . If $|B'| < |B|$, then a procedure “alphabet reduction” should take us via the compression defined by the inclusion $B' \subseteq B$ to the state $E' = (W, B', \mathcal{X}, \theta', \mu')$. Since $|B'| < |B|$, this implies $\|E'\| < \|E\|$.

We call the procedure for a given an entire solution (α, σ) only if there are no typed variables. Hence, $\theta \subseteq B^+ \times B^+$ and $\theta \neq \emptyset$ is possible. The first type is when $\sigma(X) \in M(B', \theta', \mu') \subseteq M(B, \theta, \mu)$ for all $X \in \mathcal{X}$. Indeed, as we have $W \in M(B', \mathcal{X}, \theta, \mu)$ we can define $W' = W$ and $\sigma' = \sigma$. Let $h : M(B', \mathcal{X}, \theta', \mu') \rightarrow M(B, \mathcal{X}, \theta, \mu)$ be defined by the identity on $B' \cup \mathcal{X}$. This defines a compression according to Section 5.2.2 because there are no typed variables. The corresponding transition is $(W, B, \mathcal{X}, \theta, \mu) \xrightarrow{\varepsilon} (W', B', \mathcal{X}, \theta', \mu')$. We can apply Lemma 9 and we obtain $\alpha\sigma(W) = \alpha\varepsilon\sigma'(W)$ as desired.

The second type is more subtle and we can apply it, only if $\theta = \emptyset$. Thus, we start (and end) at a standard state.

We do not require that $\sigma(X) \in M(B', \emptyset, \mu')$ as this is far from being true, in general. ($W \in M(B', \mathcal{X}, \theta, \mu)$ doesn't exclude that some $b \in B \setminus B'$ occurs in $\sigma(W)$, in general.) It is here where the notion of entire solution becomes important. We have $\alpha : M(B, \emptyset, \mu) \rightarrow M(A, \emptyset, \mu_0)$, so we can define a B' -morphism $\beta : M(B, \emptyset, \mu) \rightarrow M(A, \emptyset, \mu_0)$ by $\beta(b) = \alpha(b)$ for $b \in B \setminus B'$. Since $M(B, \emptyset, \mu) = B^*$ is a free monoid, we don't have to worry to check defining relations. Moreover, $\sigma' = \beta\sigma$ is solution at $E' = (W, B', \mathcal{X}, \emptyset, \mu')$. Thus, we can switch from (E, σ)

to $(E, \beta\sigma)$ via the transition $(W, B, \mathcal{X}, \emptyset, \mu) \xrightarrow{\varepsilon} (W, B', \mathcal{X}, \emptyset, \mu')$. Since α is an A -morphism we obtain $\alpha = \alpha\beta$. Hence, again $\alpha\sigma(W) = \alpha\beta\sigma(W) = \alpha\varepsilon\sigma(W)$ as desired.

7.2. δ -periodic compression. We begin the process with some standard state $E_s = (W_s, B_s, \mathcal{X}_s, \emptyset, \mu_s)$, $\mathcal{X}_s \subseteq \mathcal{V}$, and an entire solution (α_s, σ_s) . We assume $|W_s| \geq 8\delta n$ where $n = |W_{\text{init}}|$.

We first perform some preprocessing. We substitute every variable X , one after another in some order, either by $\sigma_s(X)$ if $|\sigma_s(X)| \leq 1$ or otherwise by aX where $\sigma_s(X) = ax$ and $a \in B_s$. Next, we call the procedure “alphabet reduction”. After that all constants are G -visible. We call the resulting sets of constants and variables B_{old} and \mathcal{X}_{old} resp. and refer to their elements as *old*. We let $E = (W, B_{\text{old}}, \mathcal{X}_{\text{old}}, \emptyset, \mu)$ together with (α, σ) denote the state and entire solution after the preprocessing. So, after these steps there is a new equation W with

$$(8) \quad |W_s| < |W| \leq |W_s| + 2|W_s|_{\mathcal{X}_{\text{old}}}.$$

Since we popped out a letter, we can guarantee the following hold:

$$(9) \quad \sum_{X \in \mathcal{X}_{\text{old}}} \alpha\sigma(W) < \sum_{X \in \mathcal{X}_s} \alpha_s\sigma_s(W).$$

This will become useful later in the termination proof.

Let us consider all very long maximal δ -periodic factors $q^d q'$, written as $up^e rv$, of $\sigma_s(W_s)$ which have an occurrence with a visible position. In the description we assume that $|u| = |v| = 3\delta$, p is primitive of length at most δ and $1 \neq r \leq p$. Hence, $up^e rv$ defines the triple (p, r, e) uniquely by Lemma 6.

The idea is that at the end we arrive at a state with a solution where all occurrences of these factors are replaced by $u[r, s, \lambda]v$ where $[r, s, \lambda]$ is the notation for a single letter and $rs = p$. Here λ is a formal symbol taken from some index set Λ . For this we write the set of these very long δ -periodic words in $\sigma(W)$ as:

$$F_\Lambda = \{u_\lambda p_\lambda^{e_\lambda} r_\lambda v_\lambda \mid \lambda \in \Lambda\}.$$

If $\Lambda = \emptyset$, then we skip the rest of δ -periodic compression. Hence, in the following we assume that Λ has minimal size (i.e. $|F_\Lambda| = |\Lambda|$) and $\Lambda \neq \emptyset$. We also define a set of primitive words

$$P_\Lambda = \{p_\lambda \mid u_\lambda p_\lambda^{e_\lambda} r_\lambda v_\lambda \in F_\Lambda\}.$$

To each $u_\lambda p_\lambda^{e_\lambda} r_\lambda v_\lambda \in F_\Lambda$ we associate a new letter $[r_\lambda, s_\lambda, \lambda]$ and we notice that we can recover $w_\lambda = u_\lambda p_\lambda^{e_\lambda} r_\lambda v_\lambda$ from $[r_\lambda, s_\lambda, \lambda]$. (That is, given $[r_\lambda, s_\lambda, \lambda]$ the words $p_\lambda, r_\lambda, u_\lambda, v_\lambda$ and the exponent e_λ are all uniquely specified.) Thus, the set

$$R_\Lambda = \{[r_\lambda, s_\lambda, \lambda] \mid \lambda \in \Lambda\}$$

is in bijection with F_Λ and hence with Λ . Moreover, we have $f(w_\lambda) = w_\lambda \iff [f(r_\lambda), f(s_\lambda), \lambda] = [r_\lambda, s_\lambda, \lambda]$, and $\overline{w_\lambda} = w_\lambda \iff \overline{r_\lambda} = r_\lambda \wedge \overline{s_\lambda} = s_\lambda$.

Since each $u_\lambda p_\lambda^{e_\lambda} r_\lambda v_\lambda$ is very long (length at least 10δ) we have $p_\lambda^{e_\lambda} r_\lambda \geq 4\delta$ and $e_\lambda \geq 3$. Let $n_\mathcal{X} = \sum_{X \in \mathcal{X}} |W|_X$ denote the number of occurrences of variables in W . For at least $|\Lambda| - 2n_\mathcal{X}$ of these factors $u_\lambda p_\lambda^{e_\lambda} r_\lambda v_\lambda$ the inner part $p_\lambda^{e_\lambda} r_\lambda$ is fully visible in W because if not, then we can assign to either a prefix or suffix of some occurrence of X . Thus,

$$(10) \quad 4\delta(|\Lambda| - 2n_\mathcal{X}) \leq |W| \text{ and hence, } 2\delta|\Lambda| \leq \frac{|W|}{2} + 4\delta n_\mathcal{X} < |W|$$

by $|W_s| < |W|$ and our assumption that W_s has length at least $8\delta n$.

Next we consider fresh variables which are denoted as $[X, f(sr)]$ where $X \in \mathcal{X} \subseteq \mathcal{V}$, $f \in G$ and $rs = p_\lambda \in P_\Lambda$. These new variables will be typed. We define the action of G by $g \cdot [X, p] = [X, g(p)]$ and the involution by $\overline{[X, p]} = [\overline{X}, \overline{p}]$. The idea is $\sigma([X, p]) \in p^*$; and thus, $\sigma(\overline{[X, p]}) \in \overline{p}^*$ and $\sigma((f, [X, p])) \in f(p)^*$. Note that $(f, [X, p]) = (g, [X, p])$ if and only if $g^{-1}f(p) = p$ and hence $g^{-1}f \in G_p$ is in the known stabilizer of p .

The next routine introduces these new variables using substitution transitions. Recall that defining $\tau(X) = w$ substitutes (f, X) by $f(w)$ and simultaneously (f, \overline{X}) by $f(\overline{w}) = \overline{f(w)}$ for all $f \in G$. Moreover, each time we define τ below, then it has to be checked that this is indeed a substitution as required in Section 5.2.1. This is straightforward and therefore left to the reader.

begin routine: “insert typed variables”.

We let $\mathcal{X} = \mathcal{X}_{\text{old}}$. For each $X \in \mathcal{X}_{\text{old}}$ do the following. (Note this means we do the process once for X and once for \overline{X} .)

begin loop

- (1) If $|\sigma(X)| \leq 20\delta - 1$, then define a substitution $\tau(X) = \sigma(X)$; remove X and switch to $\tau(E, \sigma)$; and rename it as (E, σ) . From now on $|\sigma(X)| \geq 20\delta$.
- (2) Let $q^d q'$ be the longest suffix of $\sigma(X)$ such that q is primitive, $|q| \leq \delta$, and $1 \neq q' \leq q$. If $|q^d q'| \leq 3\delta$, then we do nothing. If $|q^d q'| > 3\delta$, then define p , e , and p' by $q^d q' = up^e p'$ with $|u| = 3\delta$, p is primitive, $|p| \leq \delta$, and $1 \neq p' \leq p$. Possibly, $e = 0$. If $|p^e p'| \leq 7\delta$, then define a substitution $\tau(X) = Xp^e p'$; switch to the corresponding extended equation and rename $\tau(E, \sigma)$ as (E, σ) .
- (3) Due to the previous step we may assume $|up^e p'| > 10\delta$. In particular, $up^e p'$ is very long δ -periodic, and p is conjugate to $p_\lambda \in P_\Lambda$. We can write $\sigma(X) = xp^e p'$ with $|x| \geq 3\delta$. We enlarge \mathcal{X} by typed variables $[X, sr]$ for all factorizations $p = rs$. Moreover, if we enlarge \mathcal{X} by some $[X, p]$, then we also include $[X, f(p)]$ and $[X, f(\overline{p})]$. The number of typed variables $[X, p]$ is bounded by $\mathcal{O}(|G||W|)$ because $2\delta|\Lambda| \leq |W|$ by (10). We also update θ by adding

$$\{a[X, sa] = [X, as]a \mid \exists [X, p] : p = as \wedge a \in B_{\text{old}}\}.$$

(Note that this implies $r[X, sr] = [X, rs]r$ for every factorization $p = rs$, and in particular $p[X, p] = [X, p]p$.) Define $\tau(X) = X[X, p]p^\ell p'$ where ℓ is chosen such that $5\delta < |p^\ell p'| \leq 6\delta$. (Note that we can also write, for example, $\tau(X) = Xp^\ell p'[X, sp']$ for $p's = p$ due to θ .) This fixes the word $\sigma'(X) \in B_{\text{old}}^*$ by the condition $\sigma'(X) = x$ and

$$\sigma(X) = x\sigma'([X, p])p^\ell p'$$

- (4) If it happens that $|\sigma'(X)| = 3\delta$, then we replace X by $\sigma'(X)$; and we remove the (untyped) variable X . The B_{old} -morphism $\sigma' : M(B_{\text{old}}, \mathcal{X}, \theta, \mu) \rightarrow M(B_{\text{old}}, \emptyset, \mu)$ is well-defined by adjusting μ . Rename $\tau(E, \sigma)$ as (E, σ) .

end loop

end routine

Starting at W_s until the end of the routine we have by (8) and the first step in the routine:

$$(11) \quad |W| \leq |W_s| + 20\delta n.$$

(The careful reader might notice an overestimation of the upper bound in (11) by $|W_s|_{\mathcal{X}_{\text{old}}}$.) We define a set of fresh letters as follows

$$B_{\text{new}} = \{[sr] \mid rs \in P_\Lambda\} \cup R_\Lambda.$$

These new letters will be defined (by a morphism) so that $[p]$ represents the word $p \in B_{\text{old}}^*$ and $[r, s, \lambda]$ will represent the word $rsr \in B_{\text{old}}^*$. Enlarging possibly the set, we define $\overline{[p]} = [\overline{p}]$ and $f([p]) = [f(p)]$ for all $f \in G$. Next, we let $\overline{[r, s, \lambda]} = [\overline{r}, \overline{s}, \lambda]$ and $f([r, s, \lambda]) = [f(r), f(s), \lambda]$. (These rules are easily justified since $[r, s, \lambda]$ represents the word rsr .) Note that we have $\overline{[r, s, \lambda]} = [r, s, \lambda] \iff \overline{r} = r \wedge \overline{s} = s$, and $f([r, s, \lambda]) = [r, s, \lambda] \iff f(r) = r \wedge f(s) = s$. We have then that B_{new} is closed under involution, and the group G acts on B_{new} . For each letter its stabilizer subgroup is known. Finally, we define $\mu([p]) = \mu(p)$ and $\mu([r, s, \lambda]) = \mu(rsr)$. Every $\lambda \in \Lambda$ produces at most $\mathcal{O}(\delta)$ fresh letters in B_{new} . Since $|\Lambda| \leq |W|/2\delta$, we see $|B_{\text{new}}| \in \mathcal{O}(|G||W|)$ which is good enough. We intend to switch to the extended equation $E' = (h(W), B_{\text{old}} \cup B_{\text{new}}, \mathcal{X}, \theta', \mu')$ with the solution σ' such that $\sigma = h\sigma'$. For the switch we use a compression transition defined by $h([p]) = p$ and $h([r, s, \lambda]) = rsr$ and some $\theta' \neq \emptyset$. Since $[r, s, \lambda]$ is a letter and $|rsr| \geq 2$ we can make sure that $|h(W)| < |W|$. Hence $\|E'\| < \|E\|$.

We define

$$\begin{aligned} \theta' = \theta \cup \{([r, s, \lambda][sr] = [rs][r, s, \lambda]) \mid [r, s, \lambda] \in B_{\text{new}}\} \\ \cup \{(a[sa] = [as]a) \mid a \in B_{\text{old}}\} \cup \{[p][X, p] = [X, p][p] \mid [p] \in R_\Lambda\}. \end{aligned}$$

We have $\theta' \neq \emptyset$ because $\Lambda \neq \emptyset$. The definition $h([r, s, \lambda]) = rsr$ and $h([p]) = p$ leads to a nonerasing $(B_{\text{old}} \cup \mathcal{X})$ -morphism

$$h : M(B_{\text{old}} \cup B_{\text{new}}, \mathcal{X}, \theta', \mu) \rightarrow M(B_{\text{old}}, \mathcal{X}, \theta, \mu).$$

Now that we have set up everything we need, the next lemma shows how we can follow a transition which introduces the new letters into the equation in the correct places, preserving the solution.

Lemma 13. *Let $(W_s, B_s, \mathcal{X}_s, \emptyset, \mu_s)$ with solution σ_s with $|W_s| \geq 8\delta n$. Then there exists $W' \in M(B_{\text{old}} \cup B_{\text{new}}, \mathcal{X}', \theta', \mu')$ with $|W'| < |W_s|$ such that $h(W') = W_s$. Moreover, we can define σ' such that first, $\sigma_s(X) = h\sigma'(X)$ for all X and second, if we redefine the type by $\theta'(X) = [p]$, then $\sigma' : M(B_{\text{old}} \cup B_{\text{new}}, \mathcal{X}, \theta', \mu) \rightarrow M(B_{\text{old}} \cup B_{\text{new}}, \theta', \mu)$ becomes a solution at $(W', B_{\text{old}} \cup B_{\text{new}}, \mathcal{X}, \theta', \mu')$. The switch from $E_s = (W_s, B_{\text{old}}, \mathcal{X}_s, \emptyset, \mu_s)$ to $E' = (W', B_{\text{old}} \cup B_{\text{new}}, \mathcal{X}, \theta', \mu)$ can be realized by a compression $E_s \xrightarrow{h} E'$ in the sense of Section 5.2.2 and we have $\sigma_s(W) = h\sigma(W')$.*

Proof. Consider the occurrences of all words $w_\lambda = u_\lambda p_\lambda^{e_\lambda} r_\lambda v_\lambda$ from F_Λ which appear as maximal δ -periodic factors in $\sigma_s(W_s)$. Let us color the positions of the inner factors $p_\lambda^{e_\lambda} r_\lambda$ green. We have $h([rs]^{e_\lambda-1}[r, s, \lambda]) = p^{e_\lambda} r$ for $r = r_\lambda$ and $rs = p = p_\lambda$. Thus, if we replace the positions of $p_\lambda^{e_\lambda} r_\lambda$ by the element $[p]^{e_\lambda-1}[r, s, \lambda]$, then we obtain an element $w' \in M(B_{\text{old}} \cup B_{\text{new}}, \theta', \mu)$ such that $h(w') = \sigma(W) \in B_{\text{old}}^*$. Note that the position of the letter $[r, s, \lambda]$ is *floating* since $[rs][r, s, \lambda] = [r, s, \lambda][sr]$. In particular, $[rs]^{e_\lambda-1}[r, s, \lambda] = [r, s, \lambda][sr]^{e_\lambda-1}$ and therefore

$$\overline{[rs]^{e_\lambda-1}[r, s, \lambda]} = \overline{[r, s, \lambda][sr]^{e_\lambda-1}} = \overline{[sr]^{e_\lambda-1}[\overline{r}, \overline{s}, \lambda]}.$$

Since $\overline{\sigma(W)} = \sigma(W)$, we know $w_\lambda = \overline{w_\lambda}$. Hence $[rs]^{e_\lambda-1}[r, s, \lambda] = [\overline{rs}]^{e_\lambda-1}[\overline{r}, \overline{s}, \lambda]$ and this implies $\overline{w'} = w'$. We will use w' to construct an equation W' as follows. If one of the green positions is visible in W , then the previous steps (inside “insert typed variables” where we pop letters out) ensure that there are consecutive visible green positions which are labeled by rsr . Hence we can take that occurrence in W_s to replace it by the single letter $[r, s, \lambda]$. In case of that some of the green positions are visible some other green positions might be covered by variables. However, if so then such variables are of the form $[X, p]$. They can float and we can group them together. Thus, we can define W' such that the green positions of $p_\lambda^{e_\lambda} r_\lambda$ appear inside the equation W' as

$$[X_1, rs] \cdots [X_d, rs][rs] \cdots [rs][r, s, \lambda][X_{d+1}, sr] \cdots [X_e, sr].$$

Since we started with triangular equations, we have $0 \leq e \leq 2$; but this is not essential. Due to the letter $[r, s, \lambda]$ we have $|W'| < |W|$. It is also clear how to define $(B_{\text{old}} \cup B_{\text{new}})$ -morphism σ' such that $\sigma'(W') = w'$. We have $\sigma'([X, p]) \in p^*$ and since $\overline{w'} = w'$, we see that σ' is a solution at $(W', B_{\text{old}} \cup B_{\text{new}}, \mathcal{X}, \theta', \mu')$. Finally, we see that all requirements in Section 5.2.2 are satisfied to perform the desired switch with a compression. \square

When we introduced letters $[p]$ and $[r, s, \lambda]$, these letters and the typed variables $[X, p]$ appeared over green positions corresponding to maximal very long δ -periodic factors in $\sigma(W)$. It is therefore consistent to color now all letters $[p]$ and $[r, s, \lambda]$ and variables $[X, p]$ green. For better readability we rename E' as $E = (W, B_{\text{old}} \cup B_{\text{new}}, \mathcal{X}, \theta, \mu)$ and σ' as σ . However, we keep the colors; and we can speak of maximal green intervals of W : it is an occurrence of a green factor which cannot be extended to the left or right by any green symbol.

The next routine removes typed (green) variables and reduces the alphabet of constants to $B_{\text{old}} \cup R_\Lambda$; and all these letters will be G -visible. This allows us to speak about old and new letters when we pass to “pair compression” later. The new letters come from R_Λ , only.

begin routine: “remove typed variables and some constants”.

- (1) Whenever in the routine we see $\sigma([X, p]) = 1$ we follow a substitution defined by $\tau([X, p]) = 1$ and we remove X from \mathcal{X} . we also rename the new situation as (E, σ) .
- (2) Using, if at most two substitutions $\tau([X, p]) = [X, p][p]$ for each X , we can arrange that the equation and current solution σ satisfy:
 - $\sigma([X, p]) = [p]^\ell$ and ℓ is even;
 - Maximal green intervals have (for some $p = rs$ with $q = sr$ and $\lambda \in \Lambda$) the form $[X_1, p] \cdots [X_d, p][p]^m[r, s, \lambda][X_{d+1}, q] \cdots [X_e, q]$; and if $e \geq 1$, then $m \geq 2$.
- (3) For all $[r, s, \lambda] \in R_\Lambda$ in some order do:
 - If there is a maximal green interval

$$w = [X_1, p] \cdots [X_m, p][p]^m[r, s, \lambda][X_{d+1}, q] \cdots [X_e, q]$$

where $e \geq 0$ and m is odd, define a morphism h by $h([r, s, \lambda]) = [p][r, s, \lambda]$. Thus, we can write

$$w = h([X_1, p] \cdots [X_m, p][p]^{m-1}[r, s, \lambda][X_{d+1}, q] \cdots [X_e, q]).$$

This defines a new equation W' such that $h(W') = W$ and $|W'| \leq |W|$. Thus, there is a transition

$$E = (W, B_{\text{old}} \cup B_{\text{new}}, \mathcal{X}, \theta, \mu) \xrightarrow{h} (W', B_{\text{old}} \cup B_{\text{new}}, \mathcal{X}, \theta, \mu) = E'$$

and a new solution σ' such that $\sigma = h\sigma'$: Follow the corresponding compression substitution to (E', σ') ; and after that rename it as (E, σ) .

- (4) Due to the previous steps: whenever we see a maximal green factor $w = [X_1, p] \cdots [X_d, p][p]^m[r, s, \lambda][X_{d+1}, q] \cdots [X_e, q]$, then $\sigma([X_1, p]) \in ([p][p])^*$ and m is even. Define a morphism h by $h([p]) = [p][p]$ for all $[p]$. Thus, we can write

$$w = h([X_1, p] \cdots [X_d, p][p]^{m/2}[r, s, \lambda][X_{d+1}, q] \cdots [X_e, q]).$$

- (5) Cycle through steps (1) to (3) until all maximal green factors have length 1. This means, first the typed variables $[X, p]$ and then the letters $[p]$ are all gone, and every maximal green factor consists of a letter $[r, s, \lambda]$. Let $E = (W, B_{\text{old}} \cup B_{\text{new}}, \mathcal{X}, \theta, \mu)$ be the current state with solution σ . We have $\sigma(W) \in (B_{\text{old}} \cup R_\Lambda)^*$. Moreover, $\mathcal{X} \subseteq \mathcal{V}$ since all typed variables are gone. The inclusion $R_\Lambda \subseteq B_{\text{new}}$ induces a morphism

$$h : M(B_{\text{old}} \cup R_\Lambda, \mathcal{X}, \emptyset, \mu) \rightarrow M(B_{\text{old}} \cup B_{\text{new}}, \mathcal{X}, \theta, \mu).$$

The morphism is actually the identity on letters and the image $h(M(B_{\text{old}} \cup B'_{\text{new}}, \mathcal{X}, \emptyset, \mu))$ is a free submonoid inside $M(B_{\text{old}} \cup B_{\text{new}}, \mathcal{X}, \theta, \mu)$ because every defining relation in θ uses letters which are not present in $B_{\text{old}} \cup B'_{\text{new}}$. The corresponding transition $E = (W, B_{\text{old}} \cup B_{\text{new}}, \mathcal{X}, \theta, \mu) \xrightarrow{\varepsilon} (W, B_{\text{old}} \cup B'_{\text{new}}, \mathcal{X}, \emptyset, \mu) = E'$ is the first type of an alphabet reduction according to Section 7.1. Rename to (E, σ) .

end routine.

Proposition 14. *Let $E_s = (W_s, B_s, \mathcal{X}_s, \emptyset, \mu_s)$ be the state where we started δ -periodic-compression with $|W_s| \geq 8\delta n$; and let $E_t = (W_t, B_{\text{old}} \cup B_{\text{new}}, \mathcal{X}_t, \emptyset, \mu_t)$ the standard state where we finish δ -periodic-compression, and $(W, B, \mathcal{X}, \theta, \mu)$ any state which we have seen on the path from E_s to E_t during the procedure. Then we have:*

- $|W_t| \leq |W_s| + 20\delta n$.
- $|W| \leq |W_s| + \mathcal{O}(\delta n)$.

Moreover, let $n_{\text{new}} = \sum_{b \in B_{\text{new}}} |W_t|_b$. If $n_{\text{new}} \geq 10n$, then $|W_t| < |W_s|$.

Proof. The first assertion is (11). The second assertion is based on the fact that whenever we define a function $s : \mathbb{N} \rightarrow \mathbb{N}$ with $s(0) \leq c\delta n$ and which satisfies for all k a bound

$$s(k+1) \leq qs(k) + c\delta n$$

for some reals $q < 1$ and $c \geq 1$, then $s(k) \leq \frac{c}{1-q} \cdot \delta n$ for all $k \in \mathbb{N}$. To see where the q comes from, we choose $s(0)$ to be number of letters $[p]$ at the state where they first appear. Each time we pass through a transition defined by $h([p]) = [p][p]$ we half the number of these letters; and this shows that we can define $q = 1/2$.

Finally, let $n_{\text{new}} \geq 10n$. Then there at least $8n$ visible occurrences of new letters (all of the form $[r, s, \lambda]$) in W_t which were covered by at least 4δ visible

positions in W_s ; and these intervals of length 4δ are pairwise disjoint. So, instead of $|W_t| \leq |W_s| + 20\delta n$ we have the better estimation

$$|W_t| \leq |W_s| + 20\delta n - (4\delta - 1) \cdot 8n \leq |W_s| + 20\delta n - 3\delta \cdot 8n < |W_s|.$$

□

An example demonstrating δ -periodic compression can be found in Appendix A.

8. THE PAIR COMPRESSION METHOD

Our goal in this section is to be able to compress pairs $ab \leq W$ of constants into single letters without causing any conflict or overlap with other pairs or variables that are connected via twisted equations. To be able to find enough such pairs to do this requires some work.

8.1. Positions. Consider any standard state $E = (W, B, \mathcal{X}, \emptyset, \mu)$ together with a solution σ . We need a precise notion of equivalence between positions, which we introduce now. The idea is that whenever we modify a solution at position i , then we must modify $\sigma(W)$ at all equivalent positions $j \equiv i$.

By definition of an extended equation we can write $W = U\overline{V}$ such that $\sigma(U) = \sigma(V) = a_1 \cdots a_m$ with $a_i \in B$. We can think of $a_1 \cdots a_m$ as a labeled linear order: the interval $[1, m] \subseteq \mathbb{N}$ is the set of *positions*; and position i is labeled with the letter a_i . This induces a linear ordering on the words $U, V \in (B \cup (G \times \mathcal{X}))^*$ such that every position j with label $x \in B \cup (G \times \mathcal{X})$ (from left-to-right) corresponds to unique subinterval $I(j) \subseteq [1, m]$ of length $|\sigma(x)|$. In this way: if $x \in B$, then $I(j)$ is a single position labeled by x , and if $x = (f, X) \in G \times \mathcal{X}$, then $I(j) = [l, l + |\sigma(X)| - 1]$ and its label is $f(\sigma(X))$.

The word $\sigma(W)$ uses the interval $[1, 2m]$ and there is a natural notion of duality in $[1, 2m]$ (so on positions in W or $\sigma(W)$): we write $j \leftrightarrow \bar{j}$ if $j = m - i$ and $\bar{j} = m + i + 1$ for $0 \leq i < m$; and we say that j and \bar{j} are *dual*. Clearly, if i is labeled by a , then \bar{i} has label \bar{a} .

We assume every $X \in \mathcal{X} \subseteq \mathcal{V}$ appears in W . Let us write $\mathcal{V} = \mathcal{V}_+ \cup \mathcal{V}_-$ such that $X \in \mathcal{V}_+ \iff \overline{X} \in \mathcal{V}_-$. This allows to define a unique interval $I(X)$ for $X \in \mathcal{V}_+$ which belongs to X : it is the interval $I(j)$ of the first position in W where j 's label is X . For $\overline{X} \in \mathcal{V}_-$ we choose the set of dual positions to X . Thus, $X \neq Y$ implies $I(X) \cap I(Y) = \emptyset$; and in particular: $I(X) \cap I(\overline{X}) = \emptyset$.

The next step is that we wish to identify intervals for different occurrences of the same variable. To do so, consider a local equation $u(f, X)w(g, Y)v = uZv$ (resp. $u(f, X)wv = Zv$ or $uw(g, Y)v = uZv$) and let $I(Z) = [l, r]$. With the help of a dummy variable, we have the uniform description $u(f, X)w(g, Y)v = uZv$. (Dummies are not listed in \mathcal{X} .) In W this equation corresponds to a factorization

$$W = U_1 \# u(f, X)w(g, Y)v \# U_2 \overline{U_2} \# \overline{vZ\overline{u}} \# \overline{U_1}.$$

The positions of $u, v, w, \overline{v}, \overline{u}$ are all visible in W . Since σ is a solution, the intervals occupied by $\# \sigma(u(f, X)w(g, Y)v) \#$ and $\# \sigma(uZv) \#$ are identical in $\sigma(U) = \sigma(V) = [1, m]$. There are natural left-to-right bijections φ_1, φ_2 between $I(X)$ and the first $|\sigma(X)|$ positions belonging to $I(Z)$ and between $I(Y)$ and the last $|\sigma(Y)|$ positions belonging to $I(Z)$. Another bijection φ_3 maps the positions of $\sigma((f, X)w(g, Y))$ to $I(Z)$ such that the interval $I(w)$ of visible positions labeled by $w \in B^*$ is mapped to the middle subinterval of $I(Z)$ which is not hit by $\varphi_1(I(X))$ or $\varphi_2(I(Y))$. We will regard a position $i \in \sigma(W)$ *similar* to $j \in I(Z)$ (written as $i \sim j$) if i is mapped

to j under any bijection φ_k . This can be visualized by placing the positions of $I(X) \cup I(w) \cup I(Y)$ on “top” of $I(Z)$, as we have done before, see Figure 2. We denote by \approx the equivalence relation generated by \sim .

Finally, we let $\equiv \subseteq [1, 2m] \times [1, 2m]$ be the equivalence relation generated by \approx and \leftrightarrow . Clearly, if $i \approx j \leftrightarrow \bar{j}$ and the label at position i is $a \in B$, then a is the label at j and \bar{a} is the label at \bar{j} .

u	X	a	b	\bar{Y}	v
92, 93	32, 33, 34, 35	98, 99		42, 43, 44, 45	104, 105
92, 93	10, 11, 12, 13,	14, 15,	16, 17, 18, 19		104, 105
u	Z				v

FIGURE 2. An equation $uXab\bar{Y}v = Z$ occupying positions 92, ..., 105. We assume the variables Z, X, \bar{Y} appear in the prefix of W at positions $I(Z) = [10, 19], I(X) = [32, 35]$ and $I(\bar{Y}) = [42, 45]$.

Note that we have $i \sim j$ for positions i in $\sigma(X)$ and j in $\sigma(Y)$ if and only if $\bar{i} \sim \bar{j}$ for positions \bar{i} in $\sigma(\bar{X})$ and \bar{j} in $\sigma(\bar{Y})$. This follows because for every equation $u(f, X)w(g, Y)v = uZv$ we have its dual equation $\bar{v}(g, \bar{Y})\bar{w}(f, \bar{X})\bar{u} = \bar{v}\bar{Z}\bar{u}$ by definition of extended equations. More generally, for positions i in $\sigma(X')$ and j in $\sigma(X'')$ we have

$$i \leftrightarrow \bar{i} \sim j \iff i \sim \bar{j} \leftrightarrow j.$$

Hence,

$$(12) \quad i \equiv j \iff \text{either } i \approx j \text{ or } i \approx \bar{j}.$$

We extend the notation above to intervals. Let $p \in \mathbb{N}$. We directly link an interval $[i, i+p]$ of positions in $\sigma(X)$ (resp. $w, \sigma(Y)$) to $[j, j+p]$ in $\sigma(Z)$ if there is an equation, for example like $u(f, X)w(g, Y)v = uZv$, such that $\sigma(X)[i, i+p]$ (resp. $w[i, i+p]$, $\sigma(Y)[i, i+p]$) sits directly above the $\sigma(Z)[j, j+p]$; and we write $[i, i+p] \sim [j, j+p]$ in this case. For each interval $[i, i+p]$ of positions in $\sigma(X)$ we also let $[i, i+p] \leftrightarrow [\bar{i}+p, \bar{i}]$. As above, we let \approx and \equiv the generated equivalence relations of \sim resp. $\sim \cup \leftrightarrow$. The general form of (12) becomes

$$(13) \quad [i, i+p] \equiv [j, j+p] \iff [i, i+p] \approx [j, j+p] \vee [i, i+p] \approx [\bar{j}+p, \bar{j}].$$

We will say two positions (or intervals) are *equivalent* if they are related by \equiv . The *distance* $d(i, j)$ between positions is as usual $d(i, j) = |j - i|$.

Lemma 15. *Let $p \in \mathbb{N}$ and σ be a solution for W and W_p and σ_p be the equation and solution which we obtain as follows. For each X we do:*

- if $|\sigma(X)| \leq 2p$, then replace X by $\sigma(X)$ and remove X from the set of variables;
- if $|\sigma(X)| > 2p$, then write $\sigma(X) = uwv$ with $|u| = |v| = p$ and replace X by uXv . Change the interval $I(X) = [l, r]$ to $I_p(X) = [l+p, r-p]$. (So, it is smaller.)
- Denote the new solution for W_p defined by that procedure by σ_p .

Let i and j positions in $\sigma_p(W_p) = \sigma(W)$ which belong to variables. This means $i, j \in \bigcup \{I_p(X) \mid X \in \mathcal{X}_p\}$. Then we have $i \sim j$ (resp. $i \leftrightarrow j$) with respect to W_p and σ_p if and only if $[i - p, i + p] \sim [j - p, j + p]$ (resp. $[i - p, i + p] \leftrightarrow [j - p, j + p]$) for W and σ .

Proof. This follows directly from the definitions. See Figure 3: the white blocks are the smaller intervals I_p . \square

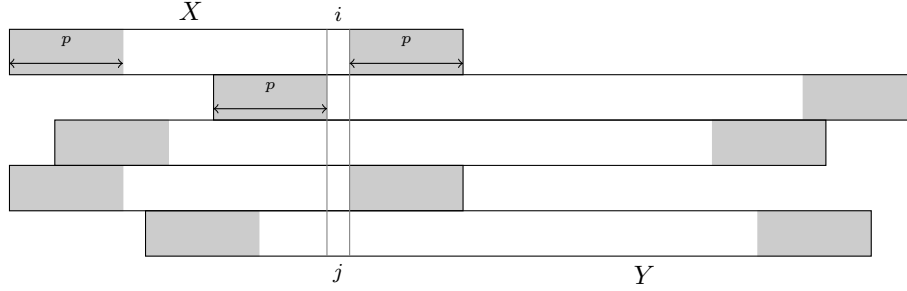


FIGURE 3. Example illustrating Lemma 15.

8.2. Red positions. We use the notation of Section 8.1. Positions at the borders of some $\sigma(X)$ inside $\sigma(W)$ play a special role because if there is a factor ab in $\sigma(W)$ and b is the (left) border of $\sigma(X)$, then we see a factor aX in W and we cannot compress ab because ab is “crossing”. The idea is to color the position of b red to signal “danger”. We color red all positions equivalent to a red position also. More precisely, consider X and the leftmost position $i \in I(X)$; and color i red. Moreover, whenever $i \equiv j$ holds in $\sigma(W)$, then color j red, too. In particular, due to the leftmost red position in $I(\bar{X})$, both the leftmost and the rightmost positions in every interval $I(X)$ are red. It follows that there are at most n pairwise different equivalence classes of red positions. This fact will be used later.

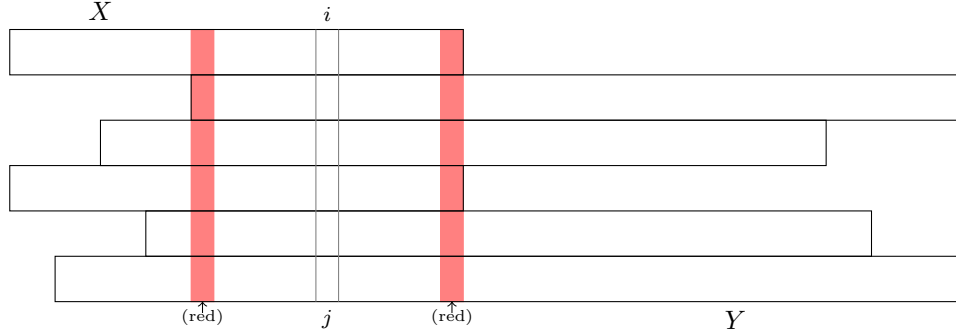


FIGURE 4. Red positions.

Lemma 16. *Let $[i-1, i, i+1, i+2]$ be an interval without any red position and where the four positions are pairwise inequivalent. Consider $[i, i+1] \equiv [j, j+1] \equiv [k, k+1]$. Then either $k = j$ and $[j, j+1] \not\approx [\bar{k}-1, \bar{k}]$ or $[j, j+1] \cap [k, k+1] = \emptyset$.*

Proof. First notice that each of the intervals $[i-1, i, i+1, i+2]$, $[j-1, j, j+1, j+2]$, $[k-1, k, k+1, k+2]$ is without red positions. By contradiction assume $j = k$ and $[j, j+1] \approx [\bar{k}-1, \bar{k}]$. Then $j \approx \bar{k}-1 \leftrightarrow k+1$ which implies $j \equiv j+1$ and hence, $i \equiv i+1$. This was excluded. Thus, for the rest of the proof we may assume, again by contradiction, $j \neq k$ and $[j, j+1] \cap [k, k+1] \neq \emptyset$. By symmetry, we assume $j+1 = k$.

We cannot have $[j, j+1] \approx [k, k+1]$ because then $j \equiv k$, but $k = j+1$ and $j \equiv j+1$ is impossible. Thus, $[j, j+1] \approx [\bar{k}-1, \bar{k}]$ and $j \equiv k+1 = j+2$. We remember $j \equiv j+2$. If $[i, i+1] \approx [j, j+1]$, then (as no position is red) $[i, i+1, i+2] \approx [j, j+1, j+2]$ implies $i \approx i+2$. This is impossible. Hence, the last option is $[\bar{j}-1, \bar{j}] \approx [i, i+1]$ and $[\bar{j}-2, \bar{j}-1, \bar{j}] \approx [i-1, i, i+1]$. However, $j \equiv j+2$ implies $\bar{j} \equiv \bar{j}-2$. We have again a contradiction as $i-1 \neq i+1$. \square

Example 17 indicates why compression is possible only in the middle interval $[i, i+1]$ of $[i-1, i, i+1, i+2]$, in general.

Example 17. *We don't exclude that G acts with involution. Thus, there might be an $a \in B$ and $f \in G$ such that $f(a) = \bar{a}$. Consider the equation $\overline{X} = (f, X)$ with the solution $\sigma(X) = dcb\bar{a}\bar{b}\bar{c}\bar{d}$ and where $f(x) = x$ for $x = b, c, d$. Then we have*

$$\sigma(\overline{X}) = dcb\bar{a}\bar{b}\bar{c}\bar{d} = f(\sigma(X)).$$

The positions of $\sigma(X)$ can be identified with $\{1, \dots, 7\}$ with $i \equiv 8-i$ for all positions $1 \leq i \leq 7$. Since positions 3 and 5 are equivalent, the interval $[2, 5]$ contains equivalent positions. The four positions in the interval $[1, 4]$ are pairwise inequivalent. However, we cannot compress the interval $[3, 4]$ corresponding to the pair ba . The reason is that $[3, 4]$ intersects with $[4, 5] = [4, \bar{3}]$. On the other hand, there is no obstacle to compress the interval $[2, 3]$.

8.3. The procedure. We define and fix $\delta = |G|\varepsilon$ and $\varepsilon = 30n$. Thus, $\delta \in \mathcal{O}(|G|n)$ and $\varepsilon \in \mathcal{O}(n)$. We start at a standard state $E = (W, B, \mathcal{X}, \emptyset, \mu)$ together with a solution σ . All local equations have the form $u(f, X)w(g, Y)v = uZv$. (As before dummy variables are allowed.)

We perform the following steps.

- (1) For every X in some order do: if $|\sigma(X)| \leq 20\delta$, then replace X by $\tau(X) = \sigma(X)$; remove X from the set of variables; rename $\tau(E, \sigma)$ as (E, σ) .
- (2) For every X in some order do: write $\sigma(X) = ux$ with $|u| = 10\delta$; replace X by $\tau(X) = uX$; rename $\tau(E, \sigma)$ as (E, σ) .

After the preceding step, define the intervals $I(X)$ as done in Section 8.1 and use a *red* color for the first position in each $I(X)$, see Section 8.2. Color in red all equivalent positions in $\sigma(W)$ of red positions, too.

Perform the following loop as long as it is possible to find an interval $[i-1, i, i+1, i+2]$ without any red position and where at least one position is visible. (Hence, all four positions are visible, since none are red).

begin loop

- (a) Let ab the label of the middle interval $[i, i+1]$. Choose fresh letter c and define a morphism h by $h(c) = ab$. Whenever $[i, i+1] \approx [j, j+1]$,

then the label of $[j, j+1]$ is $f(ab)$ for some $f \in G$. Replace each of the intervals $[j, j+1]$ and $[\bar{j}-1, \bar{j}]$ by a single new position and label this position with $f(c)$ and $f(\bar{c})$ resp. There is no conflict in this relabeling by Lemma 16. Since there is no red position, there is no “crossing” of the intervals $[j, j+1]$ or $[\bar{j}-1, \bar{j}]$. So, this gives a new but shorter equation W' . We have $h(W') = W$ and new solution σ' such that $h\sigma'(W') = \sigma(W)$. There is a new numbering for the positions, but the red positions can still be identified. The same holds for the \approx and \equiv .

- (b) Define $B' = B \cup \{f(c), f(\bar{c}) \mid f \in G\}$ and $E' = (W', B', \mathcal{X}, \emptyset, \mu')$ together with a solution σ' . There is a compression transition $E \xrightarrow{h} E'$ such that $h(E', \sigma') = (E, \sigma)$.
- (c) Rename (E', σ') as (E, σ) but transfer the induced coloring of red positions.

end loop

If we started the procedure with W and $B = B_{\text{old}} \cup B_{\text{new}}$, then at the end we obtain a system W' and B' with a solution σ' . The set of old letters B_{old} did not change, but the number of new letters is at most $|G||W'|$. It is also clear that $|W'| \leq |W| + \mathcal{O}(\delta n)$ since any increase of length is due to the first steps, where we replaced each variable X either by $\sigma(X)$ or by uXv . The worst case for $|W'|$ is that no compression took place.

We are interested to find a sufficient condition that ensures the length of W' shrinks w.r.t. $|W|$, that is, we have many pairs in W that can be compressed.

Definition 18. *The shrinking pair condition at $E = (W, B, \mathcal{X}, \emptyset, \mu)$ with a solution σ is defined by the following properties.*

- *There is a partition $B = B_{\text{old}} \cup B_{\text{new}}$ into old and new letters, which respects the involution and action by G .*
- *The number of visible positions in W labeled by new letters is at most $10n$.*
- *For no X the word $\sigma(X)$ starts with a very long δ -periodic word over old letters.*

Note this is exactly the situation we find ourselves in at the conclusion of the δ -periodic compression procedure (if W is sufficiently long).

Proposition 19. *Let (E, σ) with equation W satisfy the shrinking pair condition. If $|W| \in 20\ell\delta n + \mathcal{O}(\delta n)$, then the pair compression procedure outputs an equation W' such that $|W'| \leq |W| + 20\delta n$ and $|W'| \leq \frac{59|W|}{60} + \mathcal{O}(\delta n)$.*

8.4. Proof of Proposition 19. We begin with a situation where $\sigma(W)$ is without any coloring. For simplicity we consider equations of the form $u(f, X)w(g, Y)v = uZv$ with $\sigma(X), \sigma(Y) > 20\delta$, only. Let the intervals $I(X), I(Y), I(Z)$ be defined as in Section 8.1. Let W_1 be the equation at (E_1, σ_1) . Here E_1 denotes the state which we reach by pair compression, just after the step where every variable X' was replaced by $u'X'v'$ with $|u'| = |v'| = 10\delta$. This notation is used in Lemma 20 and Lemma 21.

Lemma 20. *Consider a subinterval $[i, j]$ of $I(Z)$ such that $i \approx j$ in W_1 with respect to σ_1 . If all positions in $[i, j]$ are labeled by old letters, then $d(i, j) > \varepsilon$.*

Proof. The choice of $[i, j]$ defines a word in B_{old}^* of length $d(i, j) + 1$. According to Lemma 15 we have $[i - p, i + p] \approx [j - p, j + p]$ for $p = 10|G|\varepsilon = 10\delta$ in W_1 . Next,

we choose ℓ as large as possible so that $[i - \ell, i + p]$ is a subinterval of $I(Z)$ and $[i - \ell, i + p] \approx [j - \ell, j + p]$ for W_1 . By induction on the number of steps using the relation \sim , this implies that there is some interval $[r, r + p + \ell]$ in $\sigma_1(Z)$ belonging to some $\sigma_1(X')$ such that $[i - \ell, i + p] \approx [r, r + p + \ell]$ in W_1 and such that r is the first position in some $\sigma_1(X')$ (and therefore red). Assume, by contradiction, $d(i, j) \leq \varepsilon$. Then, $\sigma_1(Z)[i - \ell, i + p]$ and $\sigma_1(Z)[j - \ell, j + p]$ are twisted conjugate with a positive offset $d(i, j)$ which is at most ε . This implies that $\sigma_1(Z)[i - \ell, i + p]$ is a very long δ -periodic word by Corollary 5. By hypotheses, the factor $\sigma_1(Z)[i, j]$ does not use any new letter. Hence, again by Corollary 5, the δ -periodic word $\sigma_1(Z)[i - \ell, i + p]$ is a word in old letters. Therefore, the δ -periodic prefix $\sigma_1(X')[r, r + p + \ell]$ is a word in old letters. But this contradicts the hypotheses that for no variable X' the word $\sigma(X')$ starts with a very long δ -periodic word over old letters. \square

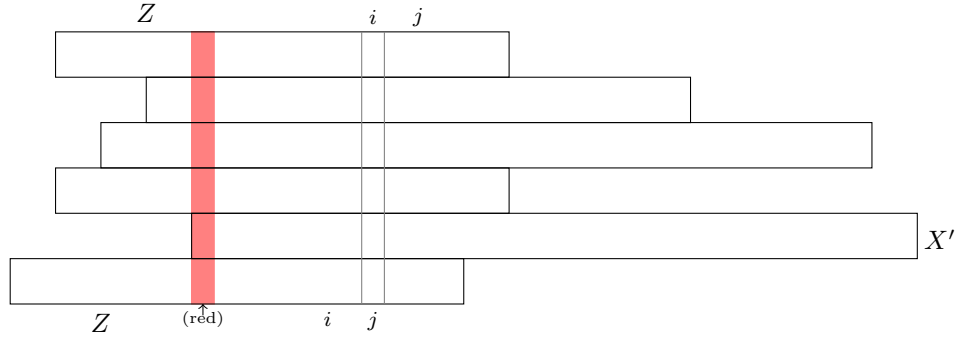


FIGURE 5. Illustration for the proof of Lemma 20.

Lemma 21. *If there are $i < j < k$ with $i \equiv j \equiv k$, then we have $d(i, k) > \varepsilon$.*

Proof. By Lemma 20 the assertion is clear if $i \approx j$ or $j \approx k$. Hence, $i \not\approx j$ so $i \approx \bar{j}$ by (12), and similarly $j \not\approx k$ so $\bar{j} \approx k$, so $i \approx k$ and hence the result by Lemma 20. \square

For the proof of Proposition 19 recall that we have $\delta = |G|\varepsilon$, $\varepsilon = 30n$, and $|W| \in 20\ell\delta n + \mathcal{O}(\delta n)$. Making the constant in the \mathcal{O} -term larger (and thereby ℓ smaller by $\mathcal{O}(1)$), we may assume that $20\ell n$ counts the number of disjoint intervals over constants in W where each interval has length exactly δ . There are at most $10n$ new letters visible in W by the shrinking pair condition. Let us call an interval *good* if it does not contain any new letter. Thus, there are at least $10\ell n$ disjoint intervals over constants in W where each interval is good and has length exactly δ . Since $\delta = |G|\varepsilon$ there are $10\ell|G|n$ disjoint good intervals of length exactly ε . Due to Lemma 21 we cannot have $i \equiv j \equiv k$ inside a good interval of length ε . Hence, there at most $2n$ red positions inside such an interval. Let us call a good interval *very good* if doesn't contain any red letter, see Section 8.2. Each of these good intervals can be subdivided into disjoint intervals of length 10. Letting $\varepsilon = 30n$, we see that there are at least $10n$ very good intervals of length 10. We claim that inside such an interval I there is an interval of length 4 where all four positions are pairwise inequivalent. Indeed, for simplicity assume $I = [1, \dots, 10]$. If all positions in $[4, 5, 6, 7]$ are pairwise inequivalent, we are done. In the other case there are at

most two equivalent positions, say $4 \equiv 6$. We cannot have $4 \approx 6$: therefore, $4 \leftrightarrow 6$. Since no position in I is red, we can conclude that $3 \leftrightarrow 7, \dots, 1 \leftrightarrow 9$. So, positions $1, \dots, 4$ found a different equivalent partner, implying that $1, \dots, 4$ are pairwise inequivalent. The other cases are similar.

Pair compression will reduce each very good interval of length 4 to length at most 3. This reduces each interval of length 10 to length at most 9. This means that a good interval of length $30n$ is reduced to the length at most $29n$. Going back to the $20\ell n$ intervals of length δ we have compression rate $\frac{59}{60}$ (or better). The fraction $\frac{59}{60}$ carries over, so after compression to all $20\ell n$ intervals, so this length reduces to at most $\frac{59}{60}20\ell\delta n$, but we must remember that ℓ was reduced by $\mathcal{O}(1)$. Using the original ℓ we have

$$(14) \quad 20(\ell - \mathcal{O}(1))\delta n \leq |W|.$$

The first step in pair compression substituted variables X by $\tau(X)$ with $|\tau(X)| \leq 1 + 20\delta$, so the initial increase is bounded by $20\delta n$. Thus, if W' denotes the equation after pair compression, then by (14) we obtain

$$(15) \quad |W'| = \frac{59}{60}20(\ell - \mathcal{O}(1))\delta n + \mathcal{O}(\delta n) \leq \frac{59}{60}|W| + \mathcal{O}(\delta n).$$

This finishes the proof of Proposition 19.

9. PUTTING IT ALL TOGETHER: THE OVERALL COMPRESSION METHOD

Now we explain what we do if we start with at the initial state E_{init} with a given initial entire solution $(\text{id}_{A^*}, \sigma_{\text{init}})$.

begin compression

Rename E_{init} and $(\text{id}_{A^*}, \sigma_{\text{init}})$ as E and (α, σ) such that $E = (W, B, \mathcal{X}, \emptyset, \mu)$. Repeat the following loop until $\mathcal{X} = \emptyset$.

begin loop

- (1) Pop out letters from variables until $|W| \geq 10\delta n$.
- (2) Let $\kappa\delta n = |W|$. Call δ -periodic-compression, and let W' denote the equation at the end of the procedure.
- (3) If $|W'| \leq \kappa\delta n$, then do nothing, else call pair-compression.

end loop

Follow a transition to a final state.

end compression

We combined the other compression methods in such a way that the intermediate equations have length in $\mathcal{O}(|G|n^2)$ and the extended alphabet has size in $\mathcal{O}(|G|^2n^2)$. We never have to store the whole alphabet, but for every symbol x occurring in the equation, we need an annotation about its stabilizer $G_x = \{f \in G \mid f(x) = x\}$. Therefore, as an upper bound we will end in $\text{NSPACE}(|G|n^2 \log(|G|n))$.

Lemma 22. *Let q, c, d reals such that $0 \leq q < 1$ and $m, m' \in \mathbb{N}$ such that*

$$m' \leq q(m + c) + d.$$

Then there is some constant κ such that $\kappa \leq m$ implies $m' < m$.

Proof. For $\kappa = \frac{c+d}{1-q}$ and $\kappa \leq m$ we have $q(m + c) + d < m$. □

The termination of the compression procedure is based the following fact. Define

$$\|E, \alpha, \sigma\| = \left(\sum_{X \text{ has no type}} |\alpha\sigma(X)|, \sum_{X \text{ is typed}} |\alpha\sigma(X)|, |W|, |B| \right)$$

to be the weight at a state with entire solution (α, σ) . Then, in the lexicographic order in \mathbb{N}^4 following transitions decreases the weight, unless we pass through the last transition to a final state. Thus, the compression procedure must stop. Moreover, if we make the space bound large enough, then we reach a final state (otherwise we could be forced to stop in the middle because we run out of space). The problem reduces to compute what it means “large enough”.

We therefore perform various rounds of the compression procedure: each round begins at a standard state E_r with equation W_r and entire solution (α_r, σ_r) . We let $(E_0, \sigma_0) = (E_{\text{init}}, \sigma_{\text{init}})$ and κ_0 some constant, say $\kappa_0 = 40$. Let κ_r be the current constant when we start round r of .

In round r , if we do not reach a final state, then there is a first time that directly after having performed a δ -periodic-compression we meet an equation W'_r such that $|W'_r| > \kappa_r \delta n$. This implies that all intermediate equations W until this point satisfied $|W| < (\kappa_r + c_r) \delta n$ for some constant c_r . Moreover, by the description of δ -periodic-compression and by Proposition 14 we know first, the number of new letters is less than $10n$ and second, we enter pair-compression. This guarantees that we satisfy the shrinking pair condition. Thus, we can define a new $\kappa_{r+1} = \kappa_r + c_r + 20 \in \kappa_r + \mathcal{O}(1)$ such that $|W'_r| \leq \kappa_{r+1} \delta n$. During pair compression the maximal increase is bounded by $20\delta n$, so we can guarantee that we don't run out of space until the end of pair-compression. Pair-compression which ends in a standard state E_{r+1} with entire solution $(\alpha_{r+1}, \sigma_{r+1})$. This finishes round r and leads us to a (E_{r+1}, σ_{r+1}) and κ_{r+1} . All the constants in the \mathcal{O} -terms are effective. This means, using Lemma 22 we can calculate a constant κ such that $\kappa_r \leq \kappa$ for all $r \in \mathbb{N}$. This proves soundness of the NFA \mathcal{A} , hence (7). Together (7) and (6) imply:

$$(16) \quad \text{Sol}(\mathcal{S}) = \{(h(c_1), \dots, h(c_k)) \in C^* \times \dots \times C^* \mid h \in L(\mathcal{A})\}.$$

The proof of Theorem 2 is finished.

10. VIRTUALLY FREE GROUPS

Recall that a group V is *virtually free* if it has a free subgroup of finite index. The aim of this section is to prove the following result.

Theorem 23. *Let V be a finitely generated virtually free group. There is an $\text{NSPACE}(m^2 \|\mathcal{S}\|^2 \log(\|\mathcal{S}\|))$ algorithm:*

Input. *A system \mathcal{S} of s equations $U_i = V_i$ over V with rational constraints and in variables X_1, \dots, X_k . We let $\|\mathcal{S}\| = k + \sum_{1 \leq i \leq s} |U_i V_i|$ and m denote the sum over the number of states for the NFA's to encode constraints.*

Output. *An extended alphabet C of size $\mathcal{O}(\|\mathcal{S}\|^2)$, letters $c_X \in C$ for each variable, and a trimmed NFA \mathcal{A} accepting a rational set of A -morphisms over C^* such that*

$$\text{Sol}(\mathcal{S}) = \{(h(c_{X_1}), \dots, h(c_{X_k})) \in C^* \times \dots \times C^* \mid h \in L(\mathcal{A})\}.$$

Moreover, $\text{Sol}(\mathcal{S}) = \emptyset$ if and only if $L(\mathcal{A}) = \emptyset$; and $|\text{Sol}(\mathcal{S})| < \infty$ if and only if \mathcal{A} doesn't contain any directed cycle.

As we mentioned in Section 1 various characterizations for finitely generated virtually free group are known. The perhaps most useful one is the characterization as fundamental groups of finite graphs of finite groups [19]. Using this result, the approach in [4] shows that a finitely generated group V is virtually free if and only if it appears as subgroup of a semi-direct product of a finitely generated free group $F(S)$ with a finite group G . The semi-direct product can be chosen in a very particular way. Let Γ be a finite connected graph with edge set $S = E(\Gamma)$ and $G \leq \text{Aut}(\Gamma)$ be subgroup of its automorphism group. We view S as finite alphabet without self-involuting letters. Hence, the action on S of G induces also a length preserving action on S^* and on $F(S)$ because reduced words are mapped to reduced words. Therefore, we can define the semi-direct product $F(S) \rtimes G$. Its elements are denoted by $[x, f]$ with $x \in F(S)$ and $f \in G$. The multiplication in $F(S) \rtimes G$ becomes

$$[x, f] \cdot [y, g] = [xf(y), fg].$$

Choosing a base point \star in Γ , the free group $F(S)$ contains the fundamental group $\pi_1(\Gamma, \star)$ as a subgroup which is a rational subset, too. We say that a group V *fits to a graph-based semi-direct product* if there are Γ and G as above such that first, V is a subgroup of $F(S) \rtimes G$, and second, the kernel of the restriction to V of the projection to the second component is exactly $F = \pi_1(\Gamma, \star)$. Thus, the projection $\varphi : V \rightarrow G$ with $(x, g) \mapsto g$ induces an embedding of V/F into G . As F is finitely generated and G is finite, we conclude that every graph-based semi-direct product is finitely generated. In order to emphasize the special structure we denote $F(S) \rtimes G$ by $\Gamma \rtimes G$.

Proposition 24. *A group V is a finitely generated virtually free group if and only if V fits to a graph-based semi-direct product $\Gamma \rtimes G$.*

Proof. The group $\Gamma \rtimes G$ is virtually free and every subgroup of a virtually free group is virtually free. The other direction is shown in [4]. See Section D.1 for details. \square

Let us show how we can use Proposition 24 in order to derive Theorem 23 from Theorem 2.

The proof has three parts.

- (1) Reduction of a system of equations over V to a system of twisted word equations over S .
- (2) Handling of rational constraints by transformations on NFA's.
- (3) Projection of EDT0L languages and reduced normal forms.

10.1. Proof of Theorem 23.

10.1.1. *Reduction of a system of equations.* We follow [4] closely. Essentially, we use only the fact that G acts on S and that V is a finitely generated subgroup of $F(S) \rtimes G$. In this part we are not concerned with any complexities, so we don't need the fact that S is the edge set of a graph. In any case, rational subsets of V become rational subsets of $F(S) \rtimes G$ and V itself is a rational set in $F(S) \rtimes G$ because V is a finitely generated.

Since V is rational in $F(S) \rtimes G$, every system of equations with rational constraints over V has a canonical interpretation as a system of equations over $F(S) \rtimes G$ with rational constraints and the (additional) constraint that every solution for a

variable must be mapped to V . Keeping this in mind, we postpone the influence of constraints.

Recall that $[x, g]$ denotes elements in the semi-direct product. Accordingly, considering a system of equations over $F(S) \rtimes G$, we denote the variable in bracket notation as $[X, g_X]$ where X is a variable over $F(S)$ and g_X is a variable over the finite group G . Since G acts on $F(S)$ we also need an action of G on the set of variables. So, without restriction, we identify each X with $(1, X)$ and for each $f, g \in G$ we introduce also a variable (g, X) such that $f \cdot (g, X) = (fg, X)$. As we intend to represent elements in $F(S)$ by reduced words, we use $\bar{x} = x^{-1}$ for $x \in F(S)$, but for $g \in G$ we write g^{-1} to denote the inverse. This is allowed because $g^{-1} = \bar{g}$ in all groups. If $[X, g_X]$ is a variable, then $[X, g_X]^{-1}$ is also defined. The semi-direct product formalism implies

$$[X, g_X]^{-1} = [(g_X^{-1}, \bar{X}), g_X^{-1}].$$

Note that in the original system over $F(S) \rtimes G$, we use variables $[X, g_X]$ and $[X, g_X]^{-1}$. Replacing $[X, g_X]^{-1}$ by $[(g_X^{-1}, \bar{X}), g_X^{-1}]$ introduces variables \bar{X} ; and the set of variables over $F(S)$ is of the form $G \times \mathcal{V}$ where \mathcal{V} is a set with involution without fixed points. Thus, we actually need a triple notation for variables $[(f, X), g_Y]$. Now, as G is finite, we can split every system of equations into a finitely many systems where each g_X is a fixed constant $g \in G$. Thus, variables have the form $[(f, X), g]$, which looks nicer, because only “twisted” variables appear. Note that splitting leads to disjoint sets of solutions since, if a variable $[(1, X), g]$ occurs, then this forces $\varphi\sigma(X) = g$ for every possible solution σ . As usual, we may triangulate a system of equations and therefore we may assume that every equation in the system has the form

$$[(f, X), f'][(g, Y), g'] = [(h, Z), h'].$$

Without restriction, $f'g' = h'$ in G because otherwise there is no solution. Thus, we are reduced to consider triangular systems of twisted equations over $F(S)$ (without constants since they can be handled by constraints) where each equation has the form

$$(f, X)(f'g, Y) = (h, Z).$$

The equivalent form is:

$$(h^{-1}f, X)(h^{-1}f'g, Y) = (1, Z).$$

Thus, after renaming it is enough to deal with equations of the form

$$(17) \quad (f, X)(g, Y) = Z.$$

This is a twisted equation over $F(S)$ and, since we are interested in normal forms, we require that a solution σ maps X to a reduced word $\sigma(X) = x$, $\sigma(Y) = y$, and $\sigma(Z) = z$ in S^* . There is a maximal suffix q of $f(x)$ such that \bar{q} appears as a prefix in $g(y)$. Thus, we can write $f(x) = pq$, $g(y) = \bar{q}r$, and $z = pr$ in reduced words. For each $h \in G$ and each equation as in (17) we introduce three fresh variables (h, P) , (h, Q) , (h, R) and we rewrite (17) as a conjunction of three equations:

$$(18) \quad (f, X) = PQ \wedge (g, Y) = \bar{Q}R \wedge Z = PR.$$

In this way our system over V with rational constraints is equivalent to a disjunction of systems of twisted word equations, but of course, we must still handle the rational constraints for variables. Even if there were no constraints at all in the beginning,

we have them now by requiring that we must have $\sigma(X) \in V$ for the original variables $[X, g_X]$. The details are explained next.

10.1.2. Handling rational constraints. It is well-known that the family of rational subsets in a given virtually free group forms an effective Boolean algebra [39, 23, 4]. This extends a classical result of Benois on free groups [2]. As we will see here, essentially, her method works for virtually free groups as well, but it involves a blow-up of state spaces, and we must prove that starting with m states the blow-up is in $\mathcal{O}(m)$. Eventually, we work with Boolean $m' \times m'$ matrices with $m' \in \mathcal{O}(m)$. This explains the factor m^2 in the statement of Theorem 23.

Recall that the inclusion of V in $\Gamma \rtimes G$ and the projection onto G induce a homomorphism denoted by $\varphi : V \rightarrow G$. By assumption its kernel is the subgroup $\pi_1(\Gamma, \star)$ of $F(S)$. Making, G smaller if necessary, we may assume that φ is surjective. In order to define a natural set A of monoid generators for V we choose a spanning tree $T_\Gamma \subseteq S = E(\Gamma)$ and we let $B = S \setminus T_\Gamma$ be the set of *bridges*. As an abstract group, F is isomorphic to $F(B)$ and we define A as the disjoint union $A = B \cup G \setminus \{1\}$. We assume that the rational constraints for variables in the original system are specified by a single NFA and for every variable the constraint is given by a certain (variable specific) subsets of final states. To simplify the notation with deal with a single variable, only: $[X, g_X]$. More specifically, according to the hypothesis in Theorem 23 we may assume $|Z| = m$. Without restriction the transitions are labeled by letters $a \in A$. Let q_0 and let Z_{fin} be the set of final states for $[X, g_X]$. Let $R \subseteq V$ be the accepted language. As we explained, it is no restriction to fix the variable g_X to be a constant $g \in G$. Since we need the constraint on X and not on $[X, g]$ we add to each state q a twin q' and two transitions (q, g^{-1}, q') and (q', g, q) . We replace Z_{fin} by its twin set Z'_{fin} . This defines a new rational set $R' \subseteq V$ such that $R'g = R$. So, the actual constraint we put on X is R' . Since we may have $g = 1$ we allow henceforth ε -transitions labeled by 1. We rename the state space of the NFA as Z and final states as Z_{fin} . We construct a new automaton \mathcal{B} as follows. The new state space is $G \times Z$ with initial state $(1, q_0)$. We also say that a state (f, p) is at *level* f . The final states are $(1, q)$ for $q \in Z_{\text{fin}}$. Note that we are interested to accept a rational set in the subgroup $\psi(F)$, only. Thus, we need to start and end in level 1. For each transition (p, a, q) with $a \in A$ we introduce for each level $f \in G$ a transition $((f, p), a, (\varphi(a)f, q))$. Thus, reading $b \in B$ doesn't change the level, but reading $1 \neq g \in G$ does. For all $p, q \in Z$ and $f \in G$: there is a path $p \xrightarrow{w} q$ in \mathcal{A} if and only if there is a path $(f, p) \xrightarrow{w} (\varphi(w)f, q)$ in \mathcal{B} . In particular, as a rational set inside V we have $L(\mathcal{A}) \cap F(B) = L(\mathcal{B})$ which is exactly what we need.

Next, we need a translation from R' into the corresponding set inside $\pi_1(\Gamma, \star) \leq F(S)$. The isomorphism $\psi : F(B) \rightarrow \pi_1(\Gamma, \star)$ maps a generator b to the path $\psi(b) = T_\Gamma[\star, s(b)] b T_\Gamma[t(b), \star]$. Here $T_\Gamma[P, Q]$ means a geodesic in the spanning tree between vertices P and Q ; and $s(b)$ (resp. $t(b)$) is the source (resp. terminus) of the bridge b . We view $\psi(b)$ as reduced word in S^* . Moreover, for each $g \in G$ we let $\psi(g) = T_\Gamma[\star, g \cdot \star]$. Using ψ we replace the labels of transitions in \mathcal{B} as follows. Each transition $((f, p), b, (f, q))$ (with $b \in B$) changes its label and becomes $((f, p), f\psi(b), (f, q))$. Each transition $((f, p), g, (gf, q))$ (with $g \in G$) changes its label and becomes $((f, p), f\psi(g), (fg, q))$.

We obtain a new automaton \mathcal{C} over S^* , but the labels of transitions are actually paths in Γ . More precisely, paths in \mathcal{C} yield paths in Γ ; and accepting paths start

and end in \star . However, these paths are not reduced, in general. Therefore, we need two more steps. Using more states we can assume that every transition on \mathcal{C} is either an ε -transition or a transition labeled by a letter $a \in S$. (The blow-up in the number of states is in $\mathcal{O}(m)$ since V is fixed.) Finally, Benois' methods [2] (being closely related to Stallings foldings) works. For states s, t in \mathcal{C} let $L(s, t) \subseteq S^*$ be the regular set of words which label a path from s to t . We perform the following loop as long as possible:

- (1) If there are $a, \bar{a} \in S$ and states s, t in \mathcal{C} such that $a\bar{a} \in L(s, t)$ introduce an ε -transition $(s, 1, t)$.
- (2) Identify states s and t if there are ε paths $s \xrightarrow{\varepsilon} t$ and $t \xrightarrow{\varepsilon} s$.

The outcome is an automaton \mathcal{D} which accepts with w the reduced paths \hat{w} in S^* , too. The number of states in \mathcal{D} is not larger than the one for \mathcal{C} .

It is instructive to see what happened to the constraint $[X, g_X] \in F(B)$. The first NFA \mathcal{A} has a single state q_0 with self-loops (q_0, b, q_0) . This forces $g_X = 1$ because otherwise we cannot satisfy the constraint. The automaton \mathcal{B} has twins, but as $g = 1$ we can identify \mathcal{A} with \mathcal{B} . The NFA \mathcal{C} is essentially a flower automaton and the procedure of Benois amounts to Stallings foldings. We end-up in an NFA which looks like Γ . However we need to accept a regular subset of normal forms in S^* which has to be in canonical bijection with its image in $F(S)$. But this is easy, we just add the constraint that we accept words only which are reduced paths starting and ending in \star . Finally we encode constraints by Boolean $m' \times m'$ matrices with $m' \in \mathcal{O}(m)$ as V is fixed. This finishes the reduction to describe solution sets for a system of equation with rational constraints as solution sets in reduced normal forms for systems of twisted word equations over S with regular constraints. What is still missing is the claim that we obtain an EDT0L language over the the alphabet \mathcal{A} .

10.1.3. Projection of EDT0L languages and reduced normal forms. Applying Theorem 2 to the situation obtained by the various construction above yields a finite number of NFA's which represent the full solution set to a system of equations with rational sets over V by a disjoint union of EDT0L languages L_i in reduced words in $\pi_1(\Gamma, \star)G \subseteq S^*$. In each of the disjoint unions the variables g_X over the group G were fixed: $g_X = g \in G$. With the help of new final states we can modify the corresponding NFA's such that L_i changes to $L_i g \subseteq S^* G$. We can combine the union over the $L_i g$ into a single EDT0L language, for example, by taking the disjoint union of several NFA's and allowing several initial states. The disjoint union is an NFA \mathcal{A} accepting a rational set of endomorphisms over some extended alphabet C with $S \subseteq C$. We let I be the set of initial states. If we apply an endomorphism $h \in L(\mathcal{A})$ to a special letter c_X we know that we obtain an EDT0L language in $S^* G$. However, our alphabet of generators is $B \cup G \setminus \{1\}$ and we wish to have normal forms in reduced words over $B^* G \subseteq (B \cup G \setminus \{1\})^*$. It is here where we use the fact that we have $B \subseteq S$. Let π_B be the projection on S^* onto B^* which maps a letter $a \in B$ to a and a letter $a \in T_\Gamma$ to the empty word 1. Then $\pi_B \psi = \text{id}_{B^*}$. Note that reduced words in $\pi_1(\Gamma, \star)$ are mapped to reduced words in B^* . Thus, the final step is to introduce a new (and unique) initial state q_I and for each $q \in I$ we introduce a transition $q_I \xrightarrow{\psi} q$. The EDT0L language we are interested is therefore the homomorphic image of the EDT0L language provided by Theorem 2. We need another property of ψ : an infinite subset in $\pi_1(\Gamma, \star)$ is mapped to an infinite subset.

This follows because $u \in \pi_1(\Gamma, \star)$ is mapped to a word of length at least $|u|/|S|$. This finishes the proof of Theorem 23.

Example 25. Let $V = \mathbb{Z}/4\mathbb{Z} \star_{\mathbb{Z}/2\mathbb{Z}} \mathbb{Z}/6\mathbb{Z} = \text{SL}(2, \mathbb{Z})$, the special linear group of 2×2 matrices over \mathbb{Z} . Then the corresponding graph of groups \mathcal{G} (with \star) is a segment

$$\mathcal{G} = \star \xrightarrow{z} P \xrightarrow{y} Q.$$

The vertex groups are $G_\star = \{1\}$, $G_P = \mathbb{Z}/4\mathbb{Z} = \langle \rho \rangle$, and $G_Q = \mathbb{Z}/6\mathbb{Z} = \langle \delta \rangle$. The nontrivial edge group is $G_y = \mathbb{Z}/2\mathbb{Z} = \langle \tau \rangle$. Hence, $\rho^2 = \tau = \delta^3$ and $\tau^2 = 1$. Mapping ρ to 3 and δ to 4, we obtain a surjective homomorphism $V \rightarrow G$ where $G = \mathbb{Z}/12\mathbb{Z}$. The kernel is a free group and has index 12 in V . (It is well-known that F is equal to the commutator subgroup $[V, V]$, but we don't need that fact.) The free subgroup F is the fundamental group of the graph Γ as seen in Figure 6. Choosing b and f as bridges we have $F = F(b, f)$: a free group of rank 2. We know more, ρ stabilizes the vertices P_α and δ stabilizes the vertices Q_β . We can identify $G/G_P = \{1, \delta, \delta^2\}$ and $G/G_Q = \{1, \rho\}$. Without restriction $\delta P_\alpha = P_{\delta\alpha}$ and $\rho Q_\beta = Q_{\rho\beta}$. Now, consider, for example, the element $v = \delta\rho\delta^2\rho\delta^2\tau \in V$. It is in normal form with respect to the amalgamated product presentation of $\text{SL}(2, \mathbb{Z})$. Thus, the translation to a reduced normal in the fundamental group with respect to \mathcal{G} becomes:

$$1z \cdot 1y \cdot \delta\bar{y} \cdot \rho y \cdot \delta^2\bar{y} \cdot \rho y \cdot \delta^2\bar{y} \cdot \tau\bar{z}.$$

Indeed, first $1z \cdot 1y$ takes us from \star to Q in \mathcal{G} , then $\delta\bar{y}$ takes us back to P and there is no (Britton-)reduction since $\delta \notin \langle \tau \rangle$, etc. Let us compute the corresponding reduced path p in Γ starting at \star . In the Bass-Serre tree we obtain a sequence

$$\star \xrightarrow{1} P_1 \xrightarrow{a} Q_1 \xrightarrow{\bar{c}} P_\delta \xrightarrow{d} Q_\rho \xrightarrow{\bar{b}} P_1 \xrightarrow{a} Q_1 \xrightarrow{\bar{f}} P_{\delta^2} \xrightarrow{\bar{\delta}^5} \delta^5 \star.$$

Projection onto bridges yields the word $\hat{v} = \bar{b}\bar{f}\delta^5 \in \hat{V}$.

REFERENCES

- [1] P. R. Asveld. Controlled iteration grammars and full hyper-AFL's. *Information and Control*, 34(3):248 – 269, 1977.
- [2] M. Benoist. Parties rationnelles du groupe libre. *C. R. Acad. Sci. Paris, Sér. A*, 269:1188–1190, 1969.
- [3] L. Ciobanu, V. Diekert, and M. Elder. Solution sets for equations over free groups are EDTOL languages. *International Journal of Algebra and Computation*, 26:843–886, 2016. Conference abstract in ICALP 2015, LNCS 9135 with full version on ArXiv e-prints: abs/1502.03426.
- [4] F. Dahmani and V. Guirardel. Foliations for solving equations in groups: free, virtually free and hyperbolic groups. *J. of Topology*, 3:343–404, 2010.
- [5] V. Diekert, C. Gutiérrez, and Ch. Hagenah. The existential theory of equations with rational constraints in free groups is PSPACE-complete. *Information and Computation*, 202:105–140, 2005. Conference version in STACS 2001, LNCS 2010, 170–182, 2004.
- [6] V. Diekert, A. Jež, and W. Plandowski. Finding all solutions of equations in free groups and monoids with involution. *Information and Computation*, 251:263–286, 2016. Conference version in Proc. CSR 2014, LNCS 8476 (2014).
- [7] V. Diekert and A. Weiß. Context-Free Groups and Their Structure Trees. *International Journal of Algebra and Computation*, 23:611–642, 2013.
- [8] M. J. Dunwoody. The accessibility of finitely presented groups. *Inventiones Mathematicae*, 81(3):449–457, 1985.
- [9] A. Ehrenfeucht and G. Rozenberg. On some context free languages that are not deterministic ETOL languages. *RAIRO Theor. Inform. Appl.*, 11:273–291, 1977.

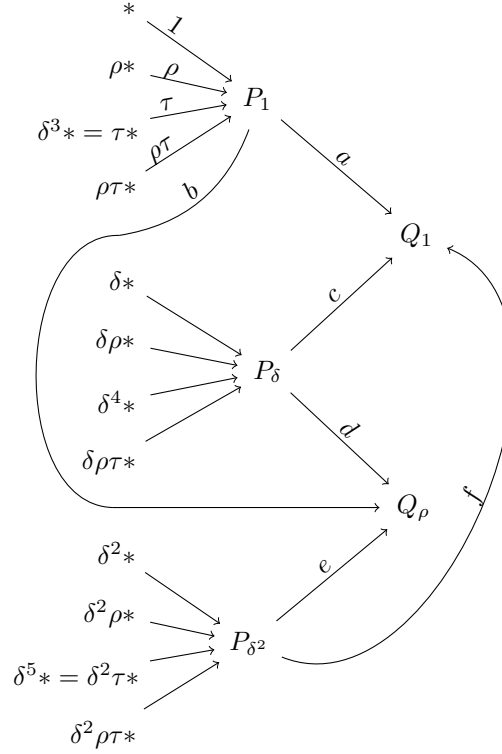


FIGURE 6. The graph Γ for $\text{SL}(2, \mathbb{Z})$ with oriented edge set $S_+ = G \cup \{a, \dots, f\}$ and brides b and f .

- [10] S. Eilenberg. *Automata, Languages, and Machines*, volume A. Academic Press, New York and London, 1974.
- [11] J. Ferté, N. Marin, and G. Sénizergues. Word-mappings of level 2. *Theory Comput. Syst.*, 54:111–148, 2014.
- [12] R. H. Gilman. Personal communication, 2012.
- [13] R. H. Gilman, S. Hermiller, D. F. Holt, and S. Rees. A characterisation of virtually free groups. *Arch. Math. (Basel)*, 89(4):289–295, 2007.
- [14] M. Gromov. Hyperbolic groups. In S. M. Gersten, editor, *Essays in Group Theory*, number 8 in MSRI Publ., pages 75–263. Springer-Verlag, 1987.
- [15] C. Gutiérrez. Satisfiability of equations in free groups is in PSPACE. In *Proceedings 32nd Annual ACM Symposium on Theory of Computing, STOC'2000*, pages 21–27. ACM Press, 2000.
- [16] M. A. Harrison. *Introduction to Formal Language Theory*. Addison-Wesley Publishing Company, 1978.
- [17] S. Jain, A. Miasnikov, and F. Stephan. The complexity of verbal languages over groups. In *Proceedings of the 27th Annual IEEE Symposium on Logic in Computer Science, LICS 2012, Dubrovnik, Croatia, June 25-28, 2012*, pages 405–414. IEEE Computer Society, 2012.
- [18] A. Jež. Recompression: a simple and powerful technique for word equations. *J. ACM*, 63(1):4:1–4:51, 2016. Conference version in Proc. STACS 2013.
- [19] A. Karrass, A. Pietrowski, and D. Solitar. Finite and infinite cyclic extensions of free groups. *Journal of the Australian Mathematical Society*, 16(04):458–466, 1973.

- [20] O. Kharlampovich and A. Myasnikov. Elementary theory of free non-abelian groups. *J. of Algebra*, 302:451–552, 2006.
- [21] D. Kozen. Lower bounds for natural proof systems. In *Proc. of the 18th Ann. Symp. on Foundations of Computer Science, FOCS’77*, pages 254–266, Providence, Rhode Island, 1977. IEEE Computer Society Press.
- [22] D. Kuske and M. Lohrey. Logical aspects of Cayley-graphs: the group case. *Ann. Pure Appl. Logic*, 131(1-3):263–286, 2005.
- [23] M. Lohrey and G. Sénizergues. Theories of HNN-extensions and amalgamated products. In M. Bugliesi, B. Preneel, V. Sassone, and I. Wegener, editors, *ICALP*, volume 4052 of *Lecture Notes in Computer Science*, pages 504–515. Springer, 2006.
- [24] G. S. Makanin. The problem of solvability of equations in a free semigroup. *Math. Sbornik*, 103:147–236, 1977. English transl. in *Math. USSR Sbornik* 32 (1977).
- [25] G. S. Makanin. Decidability of the universal and positive theories of a free group. *Izv. Akad. Nauk SSSR, Ser. Mat.* 48:735–749, 1984. In Russian; English translation in: *Math. USSR Izvestija*, 25, 75–88, 1985.
- [26] V. D. Mazurov and E. I. Khukhro, editors. *Unsolved Problems in Group Theory. The Kourovka Notebook. No. 10 (1986 edition)*. ArXiv 1401.0300, 2014.
- [27] D. E. Muller and P. E. Schupp. Groups, the theory of ends, and context-free languages. *Journal of Computer and System Sciences*, 26:295–310, 1983.
- [28] Ch. H. Papadimitriou. *Computational Complexity*. Addison Wesley, 1994.
- [29] W. Plandowski. Satisfiability of word equations with constants is in PSPACE. *J. ACM*, 51:483–496, 2004. Conference version in FOCS’99.
- [30] W. Plandowski and W. Rytter. Application of Lempel-Ziv encodings to the solution of word equations. In K. G. Larsen et al., editors, *Proc. 25th International Colloquium Automata, Languages and Programming (ICALP’98), Aalborg (Denmark), 1998*, volume 1443 of *Lecture Notes in Computer Science*, pages 731–742, Heidelberg, 1998. Springer-Verlag.
- [31] A. A. Razborov. *On Systems of Equations in Free Groups*. PhD thesis, Steklov Institute of Mathematics, 1987. In Russian.
- [32] A. A. Razborov. On systems of equations in free groups. In *Combinatorial and Geometric Group Theory*, pages 269–283. Cambridge University Press, 1994.
- [33] F. Rimlinger. Prgroups and Bass-Serre theory. *Mem. Amer. Math. Soc.*, 65(361):viii+73, 1987.
- [34] E. Rips and Z. Sela. Canonical representatives and equations in hyperbolic groups. *Inventiones Mathematicae*, 120:489–512, 1995.
- [35] G. Rozenberg and A. Salomaa. *The Book of L*. Springer, 1986.
- [36] G. Rozenberg and A. Salomaa, editors. *Handbook of Formal Languages*, volume 1. Springer, 1997.
- [37] Z. Sela. Diophantine geometry over groups VIII: Stability. *Ann. of Math.*, 177:787–868, 2013.
- [38] J.-P. Serre. *Trees*. Springer, 1980. French original 1977.
- [39] P. V. Silva. Recognizable subsets of a group: finite extensions and the abelian case. *Bulletin of the EATCS*, 77:195–215, 2002.

APPENDIX A. AN EXAMPLE FOR δ -PERIODIC COMPRESSION

Let $B_{\text{old}} = \{a, \bar{a}, b, \bar{b}\}$ of size 4. We assume that G is generated by two automorphisms f, g , both of order 2 with $f(a) = b$ and $g(a) = \bar{a}$, $g(b) = b$. (Note G is isomorphic to the dihedral group of order 8.)

Let ω, η be a large numbers, say $\omega = \eta = 1000$, and set $w = (ab)^\omega bb(ab)^\omega aabaa$. Consider a local equation

$$bba(f, X)w(g, \bar{Y})bba = bbaZbba$$

with a solution σ given as follows.

$$\begin{aligned}\sigma(X) &= (aab)^\omega (ba)^\omega \\ \sigma(Y) &= (\bar{b}aa)^\eta \bar{b} \\ \sigma(Z) &= (bba)^\omega (ab)^{2\omega} bb(ab)^\omega (aab)^{\eta+2}.\end{aligned}$$

bba	X	w	\bar{Y}	bba
	$\downarrow f$		$\downarrow g$	
bba	Z			bba

FIGURE 7. Example

Since

$$\sigma(X) = f((bba)^\omega (ab)^\omega)$$

and

$$\sigma(\bar{Y}) = b(\bar{a}\bar{a}b)^\eta = g(b(aab)^\eta)$$

we can verify that σ is indeed a solution.

For simplicity, we assume $\delta = 3$. We will follow the steps outlined in Section 7.2. We start by popping one letter left and right of every variable.

$$\underbrace{bba \overbrace{ba(bba)^{\omega-1}(ab)^{\omega-1}a}^{(f,X)} \bar{b}(ab)^\omega bb(ab)^\omega aabaa \overbrace{(aab)^{\eta-1}aa}^{(g,\bar{Y})} bbba}_Z$$

Note now we have

$$\begin{aligned}\sigma(X) &= ab(aab)^{\omega-1}(ba)^{\omega-1}b \\ \sigma(Y) &= aa(\bar{b}aa)^{\eta-1} \\ \sigma(Z) &= ba(bba)^{\omega-1}(ab)^{2\omega}bb(ab)^\omega (aab)^{\eta+1}aa.\end{aligned}$$

Next we perform alphabet reduction. Then we identify all very long (length $\geq 10\delta = 30$) maximal 3-periodic factors in the solution which have an occurrence with a position visible. We find factors $(bba)^{\omega+1}$, $(ab)^{2\omega+1}$, $(ba)^{\omega+1}$ and $(aab)^{\eta+2}$,

with the last two factors overlapping in the letter a . We write these factors as $u_\lambda p_\lambda^{\varepsilon_\lambda} r_\lambda v_\lambda$ with $|u_\lambda| = |v_\lambda| = 3\delta = 9$ to obtain the sets

$$F_\Lambda = \left\{ \begin{array}{l} (baa)^3(bba)^{\omega-6}(bba)(bba)^3, \\ (abababab)(ba)^{2\omega-9}(ba)(bababab), \\ (babababab)(ab)^{\omega-9}(ab)(abababab), \\ (aab)^3(aab)^{\eta-5}(aab)(aab)^3 \end{array} \right\},$$

$$P_\Lambda = \{bba, ba, ab, aab\}$$

and

$$R_\Lambda = \{[bba, \varepsilon, \lambda_1], [ba, \varepsilon, \lambda_2], [ab, \varepsilon, \lambda_3], [aab, \varepsilon, \lambda_4]\}.$$

(For simplicity we focus solely on the local equation.)

Next we introduce fresh typed variables following the routine “insert typed variables”. We start with X , where $\sigma(X) = ab(aab)^{\omega-1}(ba)^{\omega-1}b$.

- (1) We have $|\sigma(X)| = 5\omega - 2 \geq 20\delta$ so we skip this step.
- (2) We have $q = ba$ and $q' = b$ so that $(ba)^{\omega-1}b$ is the longest suffix of $\sigma(X)$ with q primitive and length less than 3 and $1 \neq q' \leq q$. We define $u = (ba)^4b$ of length $9 = 3\delta$, $p = ab$ and $p' = ab$ so that the suffix $(ba)^{\omega-1}b = (ba)^4b(ab)^{\omega-6}ab = up^e p'$ which has length $> 10\delta$.
- (3) Write $\sigma(X) = x(ab)^{\omega-6}ab$ where $x = ab(aab)^{\omega-1}(ba)^4b$. We add variables $[X, ab]$, $[X, ba]$ and therefore $[X, f(ab)]$, $[X, f(\overline{ab})]$, $[X, f(ba)]$, $[X, f(\overline{ba})]$ for all $f \in G$, and define

$$\theta = \{b[X, ab] = [X, ba]b, a[X, ba] = [X, ab]a, \dots\}.$$

We define $\tau(X) = X[X, ab](ab)^7ab$ since $|(ab)^7ab| = 16$ which is between 5δ and 6δ , $\sigma'(X) = ab(aab)^{\omega-1}(ba)^4b$ and $\sigma'([X, ab]) = (ab)^{\omega-13}$.

After this loop the equation now has the form

$$\underbrace{\overbrace{bbabba}^{(f,X)} \overbrace{(bba)^{\omega-1}(ab)^4a}^{(f,[X,ab])} \overbrace{(ba)^{\omega-13}(\textcolor{red}{ba})^7bab(ab)^{\omega}bb(ab)^{\omega}aabaab}^{(g,\overline{Y})} \overbrace{((aab)^{\eta-1}aa)}^{(g,\overline{Y})}}_Z bbba$$

and the solution (after renaming σ' back as σ) is

$$\begin{aligned} \sigma(X) &= ab(aab)^{\omega-1}(ba)^4b \\ \sigma([X, ab]) &= (ab)^{\omega-13} \\ \sigma(Y) &= aa(\overline{baa})^{\eta-1} \\ \sigma(Z) &= ba(bba)^{\omega-1}(ab)^{2\omega}bb(ab)^{\omega}(aab)^{\eta+1}aa. \end{aligned}$$

Next we repeat the loop for \overline{X} , where we now have $\sigma(\overline{X}) = \overline{b}(\overline{ab})^4(\overline{baa})^{\omega-1}\overline{ba}$. In this case the longest suffix $q^d q'$ is $(\overline{baa})^{\omega-1}\overline{ba} = (\overline{baa})^3(\overline{baa})^{\omega-4}\overline{ba} = up^e p'$. We introduce typed variables $[\overline{X}, \overline{baa}]$ and all variations, update θ , put $x = \overline{b}(\overline{ab})^4(\overline{baa})^3$ and define $\tau(\overline{X}) = \overline{X}[\overline{X}, \overline{baa}](\overline{baa})^5\overline{ba}$ with $\sigma'(\overline{X}) = x$ and $\sigma'([\overline{X}, \overline{baa}]) = (\overline{baa})^{\omega-6}$. After this loop the equation now has the form

$$\underbrace{\overbrace{bbab}^{(f,[\overline{X}, \overline{baa}])} \overbrace{(\textcolor{red}{ba}(bba)^5}^{(f,X)} \overbrace{(bba)^{\omega-6}}^{(f,[X,ab])} \overbrace{(bba)^3(ab)^4a}^{(f,[X,ab])} \overbrace{(ba)^{\omega-13}(\textcolor{red}{ba})^7bab(ab)^{\omega}bb(ab)^{\omega}aabaab}^{(g,\overline{Y})} \overbrace{((aab)^{\eta-1}aa)}^{(g,\overline{Y})}}_Z bbba$$

and the solution (after renaming σ' back as σ) is

$$\begin{aligned}\sigma(X) &= (aab)^3(ba)^4b \\ \sigma([X, ab]) &= (ab)^{\omega-13} \\ \sigma([\overline{X}, \overline{baa}]) &= (\overline{baa})^{\omega-6} \\ \sigma(Y) &= aa(\overline{baa})^{\eta-1} \\ \sigma(Z) &= ba(bba)^{\omega-1}(ab)^{2\omega}bb(ab)^{\omega}(aab)^{\eta+1}aa.\end{aligned}$$

APPENDIX B. RATIONAL CONSTRAINTS

B.1. Forcing homomorphisms to respect the involution. This subsection is from the arXiv version of [3]. The purpose is to define, in the language of categories, a right adjoint functor for the forgetful-functor from the category of monoids with involution to the category of monoids without involution. Our construction is fully explicit, so no knowledge about functors is needed. However, we need such a construction because regular constraints are typically defined by transition monoids over NFA's and no involution is involved here, in general.

To begin let N be any monoid. We define its dual monoid N^T to use the same set $N^T = N$, but N^T is equipped with a new multiplication $x \circ y = yx$. In order to indicate whether we view an element in the monoid N or N^T , we use a flag: for $x \in N$ we write x^T to indicate the same element in N^T . Thus, we can suppress the symbol \circ and we simply write $x^T y^T = (yx)^T$. The notation is intended to mimic transposition in matrix algebra. Similarly, we write 1 instead of 1^T which is true for the identity matrix as well. The direct product $N \times N^T$ becomes a monoid with involution by letting $\overline{(x, y^T)} = (y, x^T)$. Indeed,

$$\overline{(x_1, y_1^T) \cdot (x_2, y_2^T)} = (y_2 y_1, (x_1 x_2)^T) = \overline{(x_2, y_2^T)} \cdot \overline{(x_1, y_1^T)}.$$

The following observations are immediate.

- If N is finite then $N \times N^T$ is finite, too.
- We can embed N into $N \times N^T$ by a homomorphism $\iota : N \rightarrow N \times N^T$ defined by $\iota(x) = (x, 1)$. Note that if $\eta : N \times N^T \rightarrow N$ denotes the projection onto the first component, then $\eta\iota = \text{id}_N$.
- If M is a monoid with involution and $\rho : M \rightarrow N$ is a homomorphism of monoids, then we can lift ρ uniquely to a morphism $\widehat{\rho} : M \rightarrow N \times N^T$ of monoids with involution such that we have $\rho = \eta\widehat{\rho}$. Indeed, it is sufficient and necessary to define $\widehat{\rho}(x) = (\rho(x), \rho(\overline{x})^T)$.

Example 26 ([5]). *Let $M = \mathbb{B}^{n \times n}$. Then $M \times M^T = \mathbb{B}^{n \times n} \times (\mathbb{B}^{n \times n})^T$ is a submonoid of the set of $2n \times 2n$ -Boolean matrices:*

$$\mathbb{B}^{n \times n} \times (\mathbb{B}^{n \times n})^T = \left\{ \begin{pmatrix} P & 0 \\ 0 & Q^T \end{pmatrix} \mid P, Q \in \mathbb{B}^{n \times n} \right\} \text{ with } \overline{\begin{pmatrix} P & 0 \\ 0 & Q^T \end{pmatrix}} = \begin{pmatrix} Q & 0 \\ 0 & P^T \end{pmatrix}.$$

In the line above P^T and Q^T are the transposed matrices.

B.2. Monoids with involution and G actions. We need a right adjoint functor for the forgetful-functor from the category of monoids with involution and an action of a group G to the category of monoids (with involution). The motivation is as follows. We are working over alphabets B where some group G acts. Thus, G acts also on B^* . When we deal with a rational constraint, it is given by a morphism $\mu : B^* \rightarrow N$. Moreover, if we compress ab into a fresh letter c , then we must

compress $f(ab)$ into $f(c)$ where $f \in G$. We have to define $\mu(c) = \mu(ab)$ such that $\mu(f(c)) = f(\mu(c))$ holds. So, we need a compatible G -action on N , too. Recall that a group G acts on a set with involution M if there is a map $G \times M \rightarrow M$ such that $1 \cdot x = x$, $(fg) \cdot x = f \cdot (g \cdot x)$ and $f(\overline{x}) = \overline{f(x)}$ for all $f, g \in G$ and $x \in M$. If M is a monoid with involution, we additionally demand that $f(xy) = f(x)f(y)$. Thus, the action is given by a homomorphism from G to $\text{Aut}(M)$.

Let N be any monoid with involution. Consider the direct product N^G , which is the set of maps from G to N . We denote the elements by tuples $(n_g)_g$ with the interpretation that $g \in G$ is mapped to $n_g \in N$. It is a monoid by pointwise multiplication with involution $\overline{(n_g)_g} = (\overline{n_g})_g$. The monoid N embeds into N^G by sending n to the constant map $(n)_g$. We let G act on N^G by

$$f \cdot (n_g)_g = (n_{gf})_g.$$

Now, let G act on M and let $h : M \rightarrow N$ be a morphism of monoids with involution, then we extend it to $\hat{h} : M \rightarrow N^G$ by

$$\hat{h}(x) = (h(gx))_g.$$

The homomorphism \hat{h} respects the involution since

$$\hat{h}(\overline{x}) = (h(g\overline{x}))_g = \overline{(h(gx))_g};$$

and it respects the action of G since

$$\hat{h}(fx) = (h(gfx))_g = f \cdot \hat{h}(x).$$

Moreover, h factorizes through \hat{h} because $\hat{h}(x) = (h(gx))_g$ implies $h = \eta_1 \hat{h}$ where $\eta_1((n_g)_g) = n_1$.

Remark 27. *The application of the present subsection and Section B.1 is the following scenario. The group G is finite and acts on a (free) monoid $M = A^*$ with involution. The regular constraints are given by a homomorphism $\varphi : M \rightarrow N$ where N is finite. Then G acts on the monoid $(N \times N^T)^G$. The monoid $(N \times N^T)^G$ is a finite monoid with involution and N embeds into it by sending $n \in N$ to the constant mapping $\iota_1(n) = (n, 1)_g$; and $\eta_1((m_g, n_g^T)_g) = m_1$ defines a homomorphism such that $\eta_1 \iota_1 = \text{id}_N$. The homomorphism φ lifts uniquely to a morphism $\hat{\varphi}$ respecting the involution and the action by G such that $\eta_1 \hat{\varphi} = \varphi$.*

APPENDIX C. BASS-SERRE THEORY

For convenience of the reader we recall, mostly without any proof, some of the basic facts from Bass-Serre Theory [38]. Let Δ be a set with involution. Then and define the *free group* over Δ as

$$F(\Delta) = \Delta^* / \{a\overline{a} = 1 \mid a \in \Delta\}.$$

In case that Δ has no self-involving letters it is the standard free group of rank $|\Delta|/2$, otherwise free factors $\mathbb{Z}/2\mathbb{Z}$ appear, but this doesn't play any role here. The following text is more or less copied from [38, 7]. A *undirected graph* Γ is given by a set of vertices $V(\Gamma)$ and an edge set $E(\Gamma)$ equipped with an involution together with two mappings $s, t : E(\Gamma) \rightarrow V(\Gamma)$ where $s(y)$ denotes the *source* and $t(y)$ is the *terminus*. We require $y \neq \overline{y}$ (i.e. the involution is without fixed points), $s(\overline{y}) = t(y)$, and $t(\overline{y}) = s(y)$. An *undirected edge* is a set $e = \{y, \overline{y}\}$.

Definition 28 (Graph of Groups). Let $Y = (V(Y), E(Y))$ be a connected graph. A graph of groups \mathcal{G} over Y is given by the following data:

- (1) For all $P \in V(Y)$ there is a vertex group G_P .
- (2) For all $y \in E(Y)$ there is an edge group $G_y = G_{\bar{y}}$.
- (3) Edge groups come with an injective homomorphism from G_y to $G_{s(y)}$. It is denoted by $a \mapsto a^y$. The image of G_y in $G_{s(y)}$ is denoted by G_y^y .

Thus, for $y \in E(Y)$ with $s(y) = P$ and $t(y) = Q$ there are two isomorphisms and inclusions:

$$\begin{aligned} G_y &\xrightarrow{\sim} G_y^y \subseteq G_P, & a &\mapsto a^y, \\ G_y &\xrightarrow{\sim} G_y^{\bar{y}} \subseteq G_Q, & a &\mapsto a^{\bar{y}}. \end{aligned}$$

In order to define the fundamental group $\pi_1(\mathcal{G})$ we begin with a larger group $F_{\mathcal{G}}$. It is defined to be the free product of the free group $F(E(Y))$ with basis $E(Y)$ and the groups G_P with $P \in V(Y)$ modulo the set of defining relations $\{\bar{y}a^y y = a^{\bar{y}} \mid a \in G_y, y \in E(Y)\}$.

Thus, letting $\Sigma = E(Y) \cup \bigcup_{P \in V(Y)} (G_P \setminus \{1\})$, we have

$$F_{\mathcal{G}} = F(\Sigma) / \{gh = [gh], \bar{y}a^y y = a^{\bar{y}} \mid g, h \in G_P, a \in G_y\},$$

where $[gh]$ denotes the element obtained by multiplying g and h in G_P .

The set of paths from P to Q in Y is denoted as

$$\pi_{P,Q} = \{y_1 \cdots y_k \mid s(y_1) = P, t(y_k) = Q, t(y_i) = s(y_{i+1}) \text{ for } 1 \leq i < k\}.$$

By $\pi(\mathcal{G}, P, Q)$ we denote the subset of $F_{\mathcal{G}}$ which is defined by all $g_0 y_1 \cdots g_{k-1} y_k g_k \in F_{\mathcal{G}}$, which satisfy

$$(19) \quad y_1 \cdots y_k \in \pi_{P,Q}, g_i \in G_{s(y_{i+1})}, g_k \in G_Q \text{ for } 0 \leq i < k.$$

Note that $\pi(\mathcal{G}, P, Q) \cdot \pi(\mathcal{G}, Q, R) \subseteq \pi(\mathcal{G}, P, R)$. For $P, Q \in V(Y)$ the set $\pi(\mathcal{G}, P, P)$ is a group if and only if $P = Q$. There are two views on the fundamental group of \mathcal{G} . For this we choose a base point $\star \in V(Y)$ and a spanning tree $T \subseteq E(Y)$. (Recall that Y is connected.) We define $\pi_1(\mathcal{G}, P) = \pi(\mathcal{G}, P, P)$ and $\pi_1(\mathcal{G}, T) = F_{\mathcal{G}} / \{y = 1 \mid y \in T\}$.

Proposition 29. The canonical homomorphism φ an isomorphism from the subgroup $\pi_1(\mathcal{G}, P)$ of $F_{\mathcal{G}}$ onto the quotient group $\pi_1(\mathcal{G}, T)$.

Proof. For two vertices $Q, R \in V(Y)$ we write $T[Q, R] = y_1 \cdots y_k$ for the sequence of edges of the unique shortest path from Q to R in the spanning tree T . We can read the word $T[Q, R] \in E(Y)^*$ as an element in the group $F_{\mathcal{G}}$.

This allows us to construct a homomorphism $\psi : F_{\mathcal{G}} \rightarrow \pi_1(\mathcal{G}, P)$ by $\psi(y) = T[P, s(y)] y T[t(y), P]$ for $y \in E(Y)$ and $\psi(g) = T[P, Q] g T[Q, P]$ for $Q \in V(Y)$ and $g \in G_Q$. The homomorphism ψ is well-defined because $\psi(\bar{y}a^y y) = \psi(a^{\bar{y}})$. Moreover, ψ is surjective and $\psi(y) = 1$ for all edges $y \in E(T)$. Hence, ψ induces a surjective homomorphism of $\pi_1(\mathcal{G}, T)$ onto $\pi_1(\mathcal{G}, P)$. The composition $\pi_1(\mathcal{G}, T) \xrightarrow{\psi} \pi_1(\mathcal{G}, P) \xrightarrow{\varphi} \pi_1(\mathcal{G}, T)$ is the identity. This shows the result. \square

The group $\pi_1(\mathcal{G}) = \pi_1(\mathcal{G}, T)$ is called the *fundamental group* of the \mathcal{G} . Proposition 29 shows that the definition of a fundamental group is independent of the choice of the base point or the spanning tree. It also shows that the canonical homomorphism of G_P to $\pi_1(\mathcal{G}, T)$ is injective because $G_P \leq \pi_1(\mathcal{G}, P)$.

C.1. The Bass-Serre Tree. For each edge $y \in E(Y)$ with $s(y) = P$ fix a subset $C_y \subseteq G_P$ of representatives of the left cosets G_P/G_y^y . We require without restriction $1 \in C_y$. Thus, each $g \in G_P$ admits a unique factorization $g = ca^y$ with $c \in C_y$ and $a \in G_y$. We now define a string rewriting system $S_G \subseteq \Sigma^* \times \Sigma^*$:

$$\begin{array}{lll} gh & \longrightarrow & [gh] \quad \text{for } P \in V(Y), g, h \in G_P \setminus \{1\} \\ [ca^y]y & \longrightarrow & cya^{\overline{y}} \quad \text{for } y \in E(Y), c \in C_y, a \in G_y \setminus \{1\} \\ \overline{y}y & \longrightarrow & 1 \quad \text{for } y \in E(Y) \end{array}$$

The next proposition shows that S_G is a *convergent*. That is it is terminating and confluent; and every element has unique irreducible normal form.

Proposition 30. *We have $F_G = \Sigma^*/S_G$ and S_G is a convergent semi-Thue system. The prefix closed subset $\text{IRR}(S_G) \subseteq \Sigma^*$ of irreducible normal forms is therefore in canonical bijection with the group F_G .*

For many purposes we do not need unique normal forms, hence the rewriting system S_G is too precise and too complicated. Just as for HNN extensions we can define the notion of *Britton reductions* over \mathcal{G} . Consider an element $g_0y_1 \cdots g_{k-1}y_kg_k \in \pi(\mathcal{G}, P, Q) \subseteq F_G$ satisfying (19). Whenever we see a factor $\overline{y}a^y$ with $a \in G_y$, then we replace it by the letter $a^{\overline{y}}$; and the sequence uses less arcs. Thus, the process terminates in some *Britton-reduced* sequence which does not change the evaluation in $\pi(\mathcal{G}, P, Q)$. The following facts is crucial: if $g = g_0y_1 \cdots g_{k-1}y_kg_k$ is Britton-reduced, then applying the convergent system S_G to g yields a normal form $\hat{g} = c_0y_1 \cdots c_{k-1}y_k\tilde{g}_k$ as a word in Σ^* where $c_i \in C_{y_{i+1}}$ and $\tilde{g}_k \in G_Q$. Moreover, the y -sequence $y_1 \cdots y_k$ did not change; and the prefix $c_0y_1 \cdots c_{k-1}y_k$ depends on gG_Q , only.

We define the *Bass-Serre tree* \tilde{X} as a subset of $\text{IRR}(S_G)$. Since $\text{IRR}(S_G)$ is a prefix-closed subset of Σ^* , there is a natural tree structure given by the prefix relation on words. First, we fix a vertex $\star \in V(Y)$ as a base point. The vertices in \tilde{X} are the words $v = c_0y_1 \cdots c_{k-1}y_k \in \text{IRR}(S_G)$ such that $v \in \pi(\mathcal{G}, \star, P)$ for some $P \in V(Y)$. The root of \tilde{X} is the empty word 1. If $v = c_0y_1 \cdots c_{k-1}y_k$ is a node, then the children of v are the words vcy where $vcy \in \text{IRR}(S_G)$. We label such an arc from v to its child by cy . This means either $k = 0$ or $c \neq 1$ or $k > 0$ and $\overline{y} \neq y_k$. If $v = c_0y_1 \cdots c_{k-1}y_k$ is not the root (i.e., $k > 0$), then $c_0y_1 \cdots c_{k-2}y_{k-1}$ is the parent vertex of v . We label this arc from v to its parent by $\overline{y_k}$. The vertex $P = t(y_k)$ is uniquely defined by v . Moreover, for each edge $y = P \rightarrow Q \in E(Y)$ and each $c \in C_y$ there is a unique arc leaving v with label cy .

The map $c_0y_1 \cdots c_{k-1}y_k \mapsto c_0y_1 \cdots c_{k-1}y_k G_{t(y_k)}^y$ yields a canonical bijection between the vertices of \tilde{X} and the disjoint union $\bigcup \{\pi(\mathcal{G}, \star, P)G_P \mid P \in V(Y)\}$. The edge set of \tilde{X} corresponds to the disjoint union

$$\bigcup \{\pi(\mathcal{G}, \star, s(y))G_y^y \mid y \in E(Y)\}.$$

An arc $c_0y_1 \cdots c_{k-1}y_k \xrightarrow{cy} c_0y_1 \cdots c_{k-1}y_kcy$ of \tilde{X} is mapped to $c_0y_1 \cdots c_{k-1}y_kc G_y^y$ and $\overline{gG_y^y} = gyG_y^{\overline{y}}$. The incidences are $s(gG_y^y) = gG_{s(y)}^y$ and $t(gG_y^y) = s(gyG_y^{\overline{y}}) = gyG_{t(y)}^{\overline{y}}$, i.e., the source of an edge is defined by set wise inclusion.

Each set $\pi(\mathcal{G}, \star, P)$ is in bijection with the fundamental group $\pi_1(\mathcal{G}, T)$: to see this, choose a geodesic $\tilde{T}[P, \star]$ in \tilde{X} and map $v \in \pi(\mathcal{G}, \star, P)$ to the path $v\tilde{T}[P, \star] \in$

$\pi_1(\mathcal{G}, \star) = \pi(\mathcal{G}, T)$. This leads to another notation:

$$V(\tilde{X}) = \bigcup \{ \pi_1(\mathcal{G}, T) / G_P \cdot P \mid P \in V(Y) \}, \quad E(\tilde{X}) = \bigcup \{ \pi_1(\mathcal{G}, T) G_y \cdot y \mid y \in V(Y) \}.$$

By left multiplication we obtain a natural action of $\pi_1(\mathcal{G}, T)$ on the Bass-Serre tree \tilde{X} . Obviously the action is without edge inversion. We have $\pi_1(\mathcal{G}, T) \backslash \tilde{X} = Y$. Now, let $u \in \pi_1(\mathcal{G}, T)$ and v be any path in the Bass-Serre tree beginning at \star and ending in vG_P . We can represent u as geodesic in the Bass-Serre tree beginning at \star and ending in $u \cdot G_\star$. The path $u \cdot v$ (given by the action of $\pi_1(\mathcal{G}, T)$ on the Bass-Serre tree) is a path beginning at the node $uG_\star \in \tilde{X}$. Therefore we can compose the two paths and we obtain a new path

$$(20) \quad \star = G_\star \xrightarrow{*} u \cdot G_\star \xrightarrow{*} u \cdot vG_P.$$

Thus, the composed path has the geodesic u as a prefix, followed by a “twisted” version of v . This is why twisting comes naturally into play.

APPENDIX D. TWISTIN’ ROUND THE GRAPH

Let \mathcal{G} be a finite connected graph of finite groups and $V = \pi_1(\mathcal{G}, T)$ be its fundamental group. The section is essentially a repetition of the technique proposed by Dahmani-Guirardel in [4]. The difference is that we use an embedding of the group V as a subgroup in a semi-direct product of a free group with a finite group, whereas [4] view V as a rational subset in a free group by making the additional assumption that \mathcal{G} contains a vertex \star such that the vertex group G_\star is trivial and that \star has exactly one neighbor. We don’t need these additional assumptions, but they don’t harm neither. So, we choose any vertex \star and identify it with the root of the Bass-Serre tree. The next step is the easy direction in the proof of [19]: given a fundamental group of a finite graph of finite groups, one can construct a free normal subgroup of finite index. The construction begins with a finite set X such that $|G_P|$ divides $|X|$ for all vertex groups G_P . This allows for each G_P a free action on X . Thus, the action is without fixed points and each G_P injects (by the action) into the group of permutations $\text{Sym}(X)$. For each edge group G_y we obtain (via the vertex groups at the source of y and at the target of y) two embeddings $\alpha : G_y \rightarrow \text{Sym}(X)$ and $\beta : G_y \rightarrow \text{Sym}(X)$. We can choose systems of representatives $R, S \subseteq X$ such that $X = \bigcup_{r \in R} \alpha(G)r = \bigcup_{s \in S} \beta(G)s$ where the unions are disjoint. Because X is finite, we have $|R| = |S|$; and there is a bijection $\varphi_y : R \rightarrow S$ which can be extended an element $\varphi_y \in \text{Sym}(X)$ by $\varphi_y(\alpha(g)r) = \beta(g)s$ whenever $R \ni r \mapsto s \in S$. We obtain $\varphi_y \circ \alpha(g) \circ \varphi_y^{-1} = \beta(g)$ for all $g \in G$ because for all $\alpha(f)r \in X$ we have

$$\varphi_y \circ \alpha(g)(\alpha(f)r) = \varphi_y(\alpha(gf)r) = \beta(gf)\varphi_y(r) = \beta(g) \circ \varphi_y(\alpha(f)r).$$

By the universal property of fundamental groups of graph of groups we obtain a homomorphism $\varphi : V \rightarrow \text{Sym}(X)$. Bass-Serre theory tells us that the kernel F of φ is a free normal subgroup in V of index $|G|$ where $G = \varphi(V)$. Below we construct a basis B for F . After that we can represent elements in $w \in V$ by their reduced normal forms $\hat{w} = v\varphi(w)$ where $v \in B^*$ is the reduced word representing $w\varphi(w) \in F$. The set of reduced normal forms is a regular subset in $B^*G \subseteq (B \cup G \setminus \{1\})^*$.

Recall that V acts by left multiplication of V on the vertices $VG_P \cdot P$ and the arcs $VG_y \cdot y$ of the Bass-Serre tree. So, if we let $\Gamma = F \backslash \tilde{X}$, we obtain a finite

graph of size at most $|G| |\mathcal{G}|$. More precisely, each vertex $P \in \mathcal{G}$ has $|G/G_P|$ many copies in Γ and each arc $y \in \mathcal{G}$ has $|G/G_y|$ many copies in Γ . It is clear that G acts on the directed graph Γ as a graph morphism. Again, by Bass-Serre, F is the fundamental group of Γ . Thus, F is the set of reduced paths (paths without backtracking) in Γ from \star to \star . Let $S = E(\Gamma)$ denote the set of arcs of Γ , then F is represented a regular language in S^* . Thus, we can represent $x \in F$ as reduced word $\psi(x)$ in S^* . We obtain a basis B (closed under involution) for F as follows: choose a spanning tree $T_\Gamma \subseteq S$ and let $B = S \setminus T_\Gamma$ be set of the bridges. With each $g \in G$ we associate the unique geodesic $T[\star, g]$ in T_Γ from \star to $g \cdot \star$. After that each reduced normal form $vg \in \widehat{V}$ (where $v \in B^*$ is reduced and $g \in G$) defines a unique path as follows: for x we choose the reduced path $\psi(x) \in S^*$ from \star to \star , then we follow $T[\star, g]$ to end in $g \cdot \star$. The path is not reduced, in general. However, the reduction cannot involve any bridge since $T[\star, g]$ uses the spanning tree. Thus, each $w \in V$ defines first, a reduced normal form $\widehat{w} \in \widehat{V}$ and second, a reduced path $\psi(w)$ from \star to $\varphi(w) \cdot \star$. Vice versa, every reduced path v in Γ from \star to $g \cdot \star$ defines an element $\pi_B \in F(B)$ by projection all arcs from T_Γ to 1 and the bridges to themselves. (Thus, $F = F(B)$ is a retract in $F(S)$ since $\pi_B \psi = \text{id}_{B^*}$.) We see that starting from $w \in V$ we have

$$(21) \quad \widehat{w} = \psi(w)\varphi(g).$$

The pair $(\psi(w), \varphi(g))$ encodes w even if $G_\star \neq \{1\}$.

Remark 31. For an estimation on the size of $G = \pi_1(\mathcal{G})/F$ we have some doubly exponential bound in the size of \mathcal{G} . Indeed, appears as subgroup in $\text{Sym}(X)$ where $|X| \leq \prod \{|G_P| \mid P \in V(Y)\}$ is singly exponential in the size of \mathcal{G} .

D.1. Proof of Proposition 24. We use the same notation as above. In particular, $S = E(\Gamma)$ and $G = \varphi(V) \subseteq \text{Sym}(S)$. Recall that elements in $F(S) \rtimes G$ are denoted by $[x, f]$ with $x \in F(S)$ and $f \in G$. The multiplication in $F(S) \rtimes G$ is defined by

$$[x, f] \cdot [y, g] = [xf(y), fg].$$

Now, let $\iota : V \rightarrow F(S) \rtimes G$ be defined by

$$\iota(w) = [\psi(w), \varphi(w)].$$

Equation (21) shows that ι is injective. Let $\iota(u) = [\psi(u), f]$ and $\iota(v) = [\psi(v), g]$. It remains to show that

$$\iota(vw) = [\psi(u) f(u) \cdot v, fg].$$

The assertion $\varphi(vw) = fg$ is clear and (20) shows equality in the first component.

INSTITUT FÜR FORMALE METHODEN DER INFORMATIK, UNIVERSITÄT STUTTGART, UNIVERSITÄTSSTR. 38, D-70569 STUTTGART, GERMANY
E-mail address: diekert@fmi.uni-stuttgart.de

SCHOOL OF MATHEMATICAL AND PHYSICAL SCIENCES, THE UNIVERSITY OF NEWCASTLE, CALLAGHAN NSW 2308, AUSTRALIA
E-mail address: Murray.Elder@newcastle.edu.au